FINAL REPORT
July 31, 1998

Computational Techniques for Robust and Fixed-Structure Control Design
Grant F49620-95-1-0244

Principal Investigator

Emmanuel G. Collins
Deparment of Mechanical Engineering
Florida A&M University
2525 Pottsdamer Street
Tallahassee, FL 32310-6046
VOICE: 904-487-6373
FAX:904-487-6337
Email:ecollins@eng.fsu.edu

Period Covered: 3/1/95 - 2/28/98

Submitted To:

Dr. Marc Jacobs
Air Force Office of Scientific Research
AFOSR/NM
110 Duncan Avenue, Suite B115
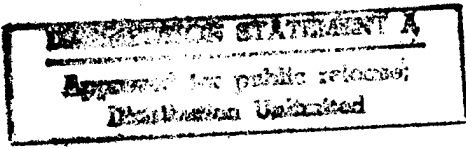Bolling Air Force Base
Washington, DC 20332-0001

1

19980824 185

REPORT DOCUMENTATION PAGE

AFRL-SR-BL-TR-98-

0582

1. AGENCY USE ONLY (Leave Blank)	2. REPORT DATE 07/31/98	3. REPORT TYPE AND DATES COVERED Final, 03/01/95---02/28/98

4. TITLE AND SUBTITLE Computational Techniques for Robust and Fixed-Structure Controller Design	5. FUNDING NUMBERS F49620-95-1-0244

6. AUTHORS
Emmanuel, G. Collins, Jr.

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Department of Mechanical Engineering FAMU/FSU College of Engineering Florida A&M University Tallahassee, FL 32310	8. PERFORMING ORGANIZATION REPORT NUMBER MECONTROL-AF-1

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) Air Force Office of Scientific Research AFOSR/NM, 110 Duncan Avenue, Suite B115 Bolling Air Force Base, Washington DC 20332-0001	10. SPONSORING / MONITORING AGENCY REPORT NUMBER :

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION / AVAILABILITY STATEMENT unlimited	12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*
In practice discrepancies between real physical systems and their mathematical models are unavoidable. These uncertainties in the mathematical model often result in severe degradation in control system performance and somtimes even instability. One of the main objectives of feedback control theory is to design controllers that are stable and guarantee certain performance objectives, in the face of these uncertainties. While robust control theory has reached a certain maturity in recent years, much remains to be done as far as numerical algorithms for practical robust controller synthesis is concerned. In practice, because of throughput limitations on control processors, the order (and sometimes additionally the structure) of the controller may have to be constrained a priori. Traditional controller reduction schemes do not guarantee robustness or optimality of the resulting controllers. The goal of this research is to develop algorithms for the analysis of closed-loop robust stability and the synthesis of fixed-structure and robust controllers.

14. SUBJECT TERMS MATHEMATICAL MODEL, FEEDBACK CONTROL THEORY, ROBUST CONTROL THEORY			15. NUMBER OF PAGES 82
			16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. Z39-1
298-102

CONTENTS

CHAPTER 1

STATUS OF EFFORT

1.1 Areas of Research

Accomplishments have been made in the following eight areas of research related to robust, fixed-structure control system design and analysis:

1. A comparison of descent and continuation techniques for H_2 optimal reduced-order control design and an investigation of the best bases in which to represent the reduced-order controller.

2. The formulation of robust, fixed-architecture control design in terms of a Riccati equation feasibility problem and the development of probability-one homotopy algorithms for its solution.

3. The implementation of probability-one homotopy algorithms for the synthesis of fixed-architecture robust controllers with H_2 or H_∞ performance and a comparison between the algorithms and controllers for H_2 and H_∞ performance using a benchmark problem.

4. The formulation of robust, fixed-architecture control design in terms of nonlinear matrix inequalities (NMI's), and the development of continuation algorithms for the solution of these NMI's.

5. The development of algorithms for the design of optimal, fixed-structure output feedback controllers for nonlinear systems.

6. The development and implementation of an object-oriented programming approach for the implementation of interior point methods to solve linear matrix inequalities (LMI's).

1

7. The development of parallel processing techniques to implement probability-one homotopy algorithms for reduced-order H_2/H_∞ control design.

8. The formulation of robustness analysis tests for discrete time systems, in the delta-domain using fixed structure multipliers.

In the chapters that follow, we motivate each of the areas of research, formulate the problems, and briefly describe key results. For further details please refer to the publications below.

1.2 Publications

1.2.1 Thesis and Dissertations

1. Sadhukhan, D., "Development of Algorithms for Synthesis of Fixed-Architecture Robust Controllers", Ph.D. Dissertation, Dept. of Mechanical Engineering, Florida A&M/Florida State University, 1998.

1.2.2 Refereed Journal Publications

1. E. G. Collins, Jr., W. M. Haddad, V. S., Chellaboina, and T. Song, "Robustness Analysis in the Delta-Domain using Fixed-Structure Multipliers," submitted to *International Journal of Robust and Nonlinear Control.*

2. E. G. Collins, Jr., W. M. Haddad, L. T. Watson, and D. Sadhukhan, "Probability-One Homotopy Algorithms for Robust Controller Synthesis with Fixed-Structure Multipliers," *International Journal of Robust and Nonlinear Control,* vol. 7, pp. 165-185, 1997.

3. E. G. Collins, Jr. and D. Sadhukhan, "A Comparison of Descent and Continuation Algorithms for H_2 Optimal Reduced-Order Control Design," *International Journal of Control,* vol. 69, pp. 647-662, 1998.

4. Y. Ge., L. T. Watson, and E. G. Collins, Jr., "Cost-Effective Parallel Processing for H_2/H_∞ Controller Synthesis," submitted to the *International Journal of Systems Science,* to appear.

5. Y. Ge, L. T. Watson, and E. G. Collins, Jr., "An Object-oriented Approach to Semidefinite Programming," *Math. Comput. Appl.*, to appear.

6. E. G. Collins, Jr., and D. Sadhukhan, "Synthesis of Fixed-Architecture, Robust H_2 and H_∞ Controllers," submitted to the *Journal of Dynamics, Systems, Measurement, and Control.*

7. E. G. Collins, Jr., and D. Sadhukhan, "Robust Controller Synthesis via Nonlinear Matrix Inequalities," submitted to the *International Journal of Control.*

8. E. G. Collins, Jr., D. Sadhukhan, and W. M. Haddad, "Fixed-Structure Nonlinear Optimal Output Feedback Stabilization for Nonlinear Systems," submitted to the *International Journal of Control.*

1.2.3 Refereed Conference Publications

1. E. G. Collins, Jr., W. M. Haddad, and L. T. Watson, "Fixed-Architecture, Robust Control Design Using Fixed-Structure Multipliers," *Proceedings of the International Federation of Automatic Control*, Vol. C, San Francisco, CA, pp. 73-78, June 1996.

2. E. G. Collins, Jr. and D. Sadhukhan, "A Comparison of Descent and Continuation Algorithms for H_2 Optimal Reduced-Order Control Design," *Proceedings of the American Control Conference*, pp. 3447-3448, 1997.

3. Y. Ge., L. T. Watson, and E. G. Collins, Jr., "A Distributed Algorithm for H_2/H_∞ Controller Synthesis," *Proceedings of the 1996 IEEE Conference on Decision and Control*, Kobe, Japan, pp. 1317-1318, Dec., 1996.

4. E. G. Collins, Jr. and D. Sadhukhan, "Synthesis of Fixed-Architecture, Robust H_2 and H_∞ Controllers," *Proceedings of the American Control Conference*, pp. 3510-3514, 1997.

5. E. G. Collins, Jr. and D. Sadhukhan, "Robust Controller Synthesis via Nonlinear Matrix Inequalities," *Proceedings of the American Control Conference*, pp. 67-71, 1997.

6. E. G. Collins, Jr., W. M. Haddad, V. S., Chellaboina, and T. Song, "Robustness Analysis in the Delta-Domain Using Fixed-Structure Multipliers," *Proceedings of the 1997 IEEE Conference on Decision and Control*, San Diego, CA, pp. 3286-3291, Dec. 1997.

7. E. G. Collins, Jr. and D. Sadhukhan, "Robust Control for a Benchmark Problem via Nonlinear Matrix Inequalities," *the 1998 American Control Conference*, to appear.

1.3   Personnel Supported

Faculty

Dr. Emmanuel G. Collins

Dr. Yuzhen Ge (Butler University)

Graduate Students

Mr. Debashis Sadhukhan

Mr. Song Tinglun

1.4   Interactions and Transitions

- Presented papers at the 1995, 1996 and 1997 IEEE Conference on Decision and Control, the 1996 International Federation of Automatic Control, and the 1997 and 1998 American Control Conference.

1.5   Honors and Awards

- Prior to grant Dr. Collins received an Honorary Superior Accomplishment Award for "Contributions in demonstrating active control of flexible spacecraft," NASA Langely Research Center, August 1991.

- During the grant Dr. Collins received The Florida State University Developing Scholar Award, April 1997.

CHAPTER 2

A COMPARISON OF DESCENT AND CONTINUATION ALGORITHMS FOR H_2 OPTIMAL, REDUCED-ORDER CONTROL DESIGN

2.1 Introduction

One of the deficiencies of modern control laws, developed by simply solving a pair of decoupled Riccati equations, in particular, globally H_2 optimal (or LQG) control and standard full-order suboptimal H_∞ control, is that the resultant control laws are always of the order of the design plant. These techniques, though relatively easy to implement computationally, do not allow the designer to constrain the architecture (e.g., order or degree of centralization) of the controller. Such constraints are often necessary in engineering practice due to throughput limitations of the control processors. Reduced-order control is therefore of paramount importance in practical control design. This chapter focuses on the design of H_2 optimal, reduced-order controllers.

Two main approaches have been developed to solve the H_2 optimal, reduced-order design problem. The first approach attempts to develop approximations to the optimal reduced-order controller by reducing the dimension of an LQG controller (Yousuff and Skelton 1984a, Yousuff and Skelton 1984b, Anderson and Liu 1989, Villemagne and Skelton 1988, Liu *et al.* 1990). These methods are attractive because they require relatively little computation and should be used if possible. Unfortunately, they tend to yield controllers that either destabilize the system or have poor performance as the requested controller dimension is decreased or the requested control authority level is increased. Hence, if used in isolation, these methods do not yield a reliable methodology for reduced-order design. In addition, these methods do not

5

extend to the design of decentralized controllers. However, it should be mentioned that, in regards to reduced-order control design, the indirect approaches at worst are valuable in providing good initial conditions for the direct approaches described below.

In contrast to controller reduction, direct approaches attempt to directly synthesize an optimal, reduced-order (or decentralized) controller by a numerical optimization scheme. There are two main classes of parameter optimization approaches to direct control design. The first class relies on the use of descent methods (Kramer and Calise 1987, Kuhn and Schmidt 1987, Kwakernaak and Sivan 1972, Ly *et al.* 1985, Mukhopadhyay 1982, Mukhopadhyay 1987, Voth and Ly 1991). Algorithms in this class reduce the H_2 cost at each iteration. For an excellent survey of descent methods as applied to output feedback problems (including methods not included in this chapter), please refer to Mäkilä and Toivonen (1987), and references therein. The second class relies on the use of continuation methods (Collins *et al.* 1995, Mercadal 1991). In contrast to the descent methods, the H_2 cost is not necessarily reduced at each iteration. It should be mentioned that continuation algorithms (Collins *et al.* 1996b) have also been developed to solve the "optimal projection equations," a set of four coupled Lyapunov and Riccati equations that characterize the H_2 optimal, reduced-order compensator. Finally, the recently developed LMI-based synthesis methods for the reduced-order control design problem (see Oliveira and Geromel (1997), and references therein), show much promise. However, these approaches will not be considered here.

From a practical design perspective it is important to determine which class of methods tends to be more numerically robust. As with the vast majority of numerical methods for nonconvex optimization problems, answers to these questions are extremely difficult to prove analytically. Instead, we must rely on numerical experimentation to observe trends. Hence, in this chapter the behavior of some standard globally convergent descent methods (i.e., steepest descent, conjugate gradient and BFGS Quasi-Newton) (Fletcher 1987) are compared to the corresponding behavior of the continuation algorithm of (Collins *et al.* 1995) by considering design for three reduced-order control design problems appearing in the literature. The Newton method is not considered here since it is not a globally convergent descent method for

nonquadratic cost functions. However when suitably modified it displays good convergence properties (Mäkilä and Toivonen 1987, Toivonen and Mäkilä 1987, Beseler *et. al.* 1992).

2.2 H_2 Optimal, Reduced-Order Dynamic Compensation

2.2.1 Problem Formulation

Consider the system

$$\dot{x}(t) = Ax(t) + Bu(t) + D_1 w(t) \tag{2.1}$$
$$y(t) = Cx(t) + Du(t) + D_2 w(t) \tag{2.2}$$
$$z(t) = E_1 x(t) + E_2 u(t) \tag{2.3}$$

where $w \in \mathcal{R}^{n_w}$ is white noise with unit intensity, $x \in \mathcal{R}^{n_x}$, $u \in \mathcal{R}^{n_u}$, $y \in \mathcal{R}^{n_y}$, $z \in \mathcal{R}^{n_z}$, D_2 has full row rank, and E_2 has full column rank. We desire to design a n_c^{th} order dynamic compensator,

$$\dot{x}_c(t) = A_c x_c(t) + B_c y(t) \tag{2.4}$$
$$u(t) = -C_c x_c(t) \tag{2.5}$$

where $n_c \leq n_x$, which minimizes the steady state performance criterion

$$J(A_c, B_c, C_c) \triangleq \lim_{t \to \infty} E[z^T(t) z(t)]. \tag{2.6}$$

The state-space evolution of the closed-loop system corresponding to (2.1)-(2.5) is described by

$$\dot{\tilde{x}}(t) = \tilde{A}\tilde{x}(t) + \tilde{D}w(t) \tag{2.7}$$

where

$$\tilde{x}(t) \triangleq \begin{bmatrix} x(t) \\ x_c(t) \end{bmatrix}, \quad \tilde{A} \triangleq \begin{bmatrix} A & -BC_c \\ B_c C & A_c - B_c D C_c \end{bmatrix}, \quad \tilde{D} \triangleq \begin{bmatrix} D_1 \\ B_c D_2 \end{bmatrix}. \tag{2.8}$$

To guarantee that the cost J is finite and independent of initial conditions we restrict our attention to the set of stabilizing compensators, $\mathcal{S}_c \triangleq \{(A_c, B_c, C_c) : \tilde{A}$

is asymptotically stable}. Assume $(A_c, B_c, C_c) \in \mathcal{S}_c$ and define $\tilde{Q} \in \mathcal{R}^{(n_x+n_c)\times(n_x+n_c)}$ to be the closed-loop steady-state covariance, i.e.,

$$0 = \tilde{A}\tilde{Q} + \tilde{Q}\tilde{A}^T + \tilde{V} \tag{2.9}$$

where

$$\tilde{V} \triangleq \begin{bmatrix} V_1 & V_{12}B_c^T \\ B_cV_{12}^T & B_cV_2B_c^T \end{bmatrix}, \quad V_1 \triangleq D_1^TD_1, \quad V_{12} \triangleq 2D_1^TD_2, \quad V_2 \triangleq D_2^TD_2. \tag{2.10}$$

(Note that since D_2 has full row rank, $V_2 > 0$.) The cost function J can now be expressed as

$$J(A_c, B_c, C_c, \tilde{Q}) = \text{tr}\tilde{Q}\tilde{R}. \tag{2.11}$$

where

$$\tilde{R} \triangleq \begin{bmatrix} R_1 & R_{12}C_c \\ C_c^TR_{12}^T & C_c^TR_2C_c \end{bmatrix}, \quad R_1 \triangleq E_1^TE_1, \quad R_{12} \triangleq -E_1^TE_2, \quad R_2 \triangleq E_2^TE_2. \tag{2.12}$$

(Note that since E_2 has full column rank, $R_2 > 0$.) The objective is to minimize the cost function J subject to the constraint (2.9).

The Lagrangian $\mathcal{L}$ is defined by

$$\mathcal{L}(A_c, B_c, C_c, \tilde{Q}, \tilde{P}) \triangleq \text{tr}\tilde{Q}\tilde{R} + \text{tr}[\tilde{P}(\tilde{A}\tilde{Q} + \tilde{Q}\tilde{A}^T + \tilde{V})] \tag{2.13}$$

where $\tilde{P}$ is the Lagrange multiplier matrix. The compensator (A_c, B_c, C_c) is optimal if it satisfies the stationary conditions

$$\frac{\partial \mathcal{L}}{\partial A_c} = 0, \quad \frac{\partial \mathcal{L}}{\partial B_c} = 0, \quad \frac{\partial \mathcal{L}}{\partial C_c} = 0, \tag{2.14}$$

$$\frac{\partial \mathcal{L}}{\partial \tilde{Q}} = \tilde{A}^T\tilde{P} + \tilde{P}\tilde{A} + \tilde{R} = 0, \tag{2.15}$$

and

$$\frac{\partial \mathcal{L}}{\partial \tilde{P}} = \tilde{A}\tilde{Q} + \tilde{Q}\tilde{A}^T + \tilde{V} = 0. \tag{2.16}$$

Both the descent and continuation algorithms aim at finding $(A_c, B_c, C_c) \in \mathcal{S}_c$ that satisfy the above conditions.

Subsequently, we will represent the controller by a parameter vector θ. When the controller is not constrained to any basis, the parameter vector θ is given by

$$\theta = \begin{bmatrix} \text{vec}(A_c) \\ \text{vec}(B_c) \\ \text{vec}(C_c) \end{bmatrix}. \tag{2.17}$$

Note that when the controller is constrained to a basis, θ contains only the *free* parameters of the controller matrices and hence, in general, is a subset of 2.17. Let the mapping from a state space representation of a controller (A_c, B_c, C_c) to the parameter vector θ be given by $g(\cdot)$, such that

$$\theta = g(A_c, B_c, C_c) \tag{2.18}$$

and define

$$\Theta = \{\theta = g(A_c, B_c, C_c) : (A_c, B_c, C_c) \in S_c \cap \text{dom}(g)\} \tag{2.19}$$

Now, assuming $\theta \in \Theta$, the H_2 cost functional and the corresponding Lagrangian can be expressed respectively as $J(\theta, \tilde{Q})$ and $\mathcal{L}(\theta, \tilde{Q}, \tilde{P})$. The problem is therefore to find $\theta \in \Theta$ such that

$$0 = \frac{\partial \mathcal{L}}{\partial \theta}(\theta, \tilde{Q}, \tilde{P}). \tag{2.20}$$

subject to (2.16) and (2.15).

2.3 Parameter Optimization Algorithms

This section first gives a general description of the algorithms corresponding to the descent methods. It then briefly describes a continuation algorithm. Particular attention is given to the modification of these algorithms to take into account the constraint $\theta \in \Theta$.

2.3.1 Descent Methods

Descent methods are designed to search for solutions to the unconstrained optimization problem

$$\min_{\theta} J(\theta). \tag{2.21}$$

The user is required to supply an initial parameter vector θ_0. A descent algorithm then has the following structure.

A Descent Algorithm

1. Let $k = 0$.

2. Determine a search direction d_k.

3. Use a one dimensional line search to find α_k that minimizes $J(\theta_k + \alpha d_k)$ with respect to α.

4. Set $\theta_{k+1} = \theta_k + \alpha_k d_k$

5. If the gradient $\frac{\partial J}{\partial \theta}(\theta_{k+1})$ is sufficiently small, then let the optimal solution $\theta^* = \theta_{k+1}$ and stop, else let $k = k + 1$ and go to Step 2.

Alternative descent methods differ primarily in the way they compute the descent direction d_k. For example, in the steepest descent method d_k corresponds to the negative of the gradient. Conjugate gradient and Quasi-Newton methods compute d_k using only cost and gradient information while Newton's method requires computation of the Hessian matrix. Note that for the H_2 optimal, reduced-order control problem it is not difficult to show that if (2.16) and (2.15) are satisfied, then the gradient satisfies

$$\frac{\partial J}{\partial \theta} = \frac{\partial \mathcal{L}}{\partial \theta}. \tag{2.22}$$

Hence, the gradient may be computed by constructing and differentiating the Lagrangian.

Recognize that the H_2 optimal, reduced-order control problem is not the unconstrained optimization problem (2.21) but is actually the constrained optimization problem

$$\min_{\theta \in \Theta} J(\theta) \tag{2.23}$$

where Θ is defined by (2.19). One way to take into account the constraint $\theta \in \Theta$ is to modify the line search subalgorithm of Step 3 to ensure that if $\theta_k \in \Theta$, θ_{k+1} is also in Θ. (It is assumed that $\theta_0 \in \Theta$).

The descent algorithms compared in this chapter use the modified line search algorithm of Kuhn and Schmidt (1987). The fundamental idea consists of decomposing the values of the line search parameter α into three regions:

1. $[0, \alpha_i^*]$: left of minimum

2. $[\alpha_i^*, \alpha_{ib})$: right of minimum, stable

3. $[\alpha_{ib}, \infty)$: unstable

where α_i^* denotes the minimum and α_{ib} denotes the stability boundary. The algorithm finds an $\alpha_{i1} \in [0, \alpha_i^*]$ and an $\alpha_{i2} \in [\alpha_i^*, \alpha_{ib})$. The minimum, α_{ih}, of an approximating cubic interpolant, is then used to subdivide the interval $[\alpha_{i1}, \alpha_{i2}]$. For the sub-interval a new α_{ih} can be found. This process is continued until no significant improvement in the approximation to the minimum step size can be achieved.

2.3.2 Continuation Methods

Continuation techniques can be used to solve the zero finding problem

$$0 = f(\theta), \qquad (2.24)$$

where $f : \mathcal{R}^p \to \mathcal{R}^p$. In the context of H_2 optimal, reduced-order control, (2.24) corresponds to (2.20). Continuation techniques work by finding a C^2 function $H : \mathcal{R}^p \times [0, 1) \to \mathcal{R}^p$ that satisfies certain properties, including the following:

1. $H(\theta, 1) = f(\theta)$;

2. $0 = H(\theta, 0)$ has an easily found or known solution θ_0.

They then trace the zero curve described by

$$0 = H(\theta, \lambda), \quad \lambda \in [0, 1). \qquad (2.25)$$

This is accomplished by differentiating (2.25) with respect to λ to obtain Davidenko's differential equation

$$0 = H_\lambda(\theta, \lambda) + H_\theta(\theta, \lambda)\theta_\lambda(\lambda) \qquad (2.26)$$

where $H_\lambda \triangleq \frac{\partial H}{\partial \lambda}$, $H_\theta \triangleq \frac{\partial H}{\partial \theta}$, and $\theta_\lambda \triangleq \frac{d\theta}{d\lambda}$, which together with $\theta(0) = \theta_0$ defines an initial value problem. Predictor-corrector, numerical integration schemes are then used to solve this initial value problem, that is to follow the curve (2.25) from the solution θ_0 of $0 = H(\theta, 0)$ to a solution θ^* of $0 = H(\theta, 1)$. In particular a continuation algorithm has the following structure.

A Continuation Algorithm

1. Let $\lambda = 0$ and $\theta(\lambda) = \theta_0$.

2. Use (2.26) to compute the tangent vector θ_λ, such that $\theta_\lambda(\lambda) = -H_\theta(\theta, \lambda)^{-1} H_\lambda(\theta, \lambda)$.

3. For some $\Delta\lambda$ such that $\lambda = \lambda + \Delta\lambda \leq 1$, use current and past values of H and H_λ to predict $\theta(\lambda + \Delta\lambda)$ by using polynomial curve fitting.

4. Let $\lambda \leftarrow \lambda + \Delta\lambda$ and θ_0 be the prediction of $\theta(\lambda)$.

5. For $k = 0, 1, 2, \ldots$ until convergence, do

$$\theta_{k+1} = \theta_k - H_\theta(\theta_k, \lambda)^{-1} \theta_k.$$

   Then, let $\theta(\lambda) = \theta_{k+1}$.

6. If $\lambda < 1$, go to Step 2, else if $\lambda = 1$, then let the solution $\theta^* = \theta(\lambda)$ and stop.

The initializing controller θ_0 in the algorithm for H_2 optimal, reduced-order control is usually found by applying a controller reduction method such as balanced controller reduction (Yousuff and Skelton 1984a) to a low authority LQG controller (Collins *et al.* 1995, Collins *et al.* 1996a) since this usually yields a nearly optimal, reduced-order controller. The initial weights $(R_1)_0$, $(R_{12})_0$, $(R_2)_0$, $(V_1)_0$, $(V_{12})_0$, $(V_2)_0$ corresponding to the low authority LQG controller are then deformed into the desired weights along the homotopy path. The reader is referred to (Collins *et al.* 1995) for further details.

The algorithm of (Collins *et al.* 1995) also assumes that the prediction $\theta(\lambda + \Delta\lambda) \in \Theta$ such that it corresponds to a controller that stabilizes the closed-loop system. If $\theta(\lambda + \Delta\lambda) \notin \Theta$, then the algorithm reduces the size of $\Delta\lambda$. In particular, $\Delta\lambda \leftarrow \frac{1}{2}\Delta\lambda$.

2.4 Numerical Examples

2.4.1 Description of Problems

The first problem is a noncollocated axial vibration control problem involving an axial beam with four circular disks attached. This problem was introduced in

(Cannon and Rosenthal 1984) and also studied in (Collins *et al.* 1995). The plant is 8th order while we consider the design of a 4th order controller.

The second problem was introduced in (Ly *et al.* 1985) and involves flight control for a NAVION aircraft. The model is 7th order and we consider the design of a 4th order controller.

The third problem was introduced in (Martin and Bryson 1980) and involves vibration control of a flexible spacecraft. The model is 6th order while we again consider the design of a 4th order controller.

It may be advantageous to keep the dimension of the optimization variable small. Hence the effect of constraining the controller to three bases: the tridiagonal form, the second order polynomial form (SPF), and the controllability canonical form (CCF) is also investigated. We design higher order controllers for all three examples using the continuation and the BFGS algorithms with the controller unconstrained and with the controller constrained to the tridiagonal basis, in order to compare the two bases.

Note that the matrices R_2 and V_2 are multiplied by ρ which is allowed to change from 10 to 1 in order to deform the low authority controller to a higher authority controller using the continuation algorithm. In the case of the descent algorithms ρ is fixed at 1.

For each example, a low authority optimal LQG controller (corresponding to $\rho = 10$) is first designed. The order of this controller is then reduced using the modified balanced controller reduction technique of (Yousuff and Skelton 1984a). This reduced order sub-optimal controller is then converted into an optimal low authority controller using a few Newton iterations. This controller is used as the starting point for both the continuation and descent optimization methods. Both the BFGS and the conjugate gradient algorithms are implemented with restarts to make them globally convergent. Convergence is said to have been achieved when the magnitude of the normalized gradient ($\triangleq \frac{\partial J}{\partial \theta} \frac{\|\theta\|}{J}$) falls below 10^{-2}.

In both the descent and the continuation algorithms, the gradient calculation is not optimized. For example, the order of operations in triple matrix products is not optimized and the fact that the Lyapunov equations (2.15) and (2.16) are transposes

Method	Basis	Function	Gradient	Hessian	Mflops	sec
Continuation	Unconstrained	58	58	58	68.8	36.7
	Tridiagonal	60	60	60	51	31.5
	SPF	351	351	351	178.8	110.7
	CCF	274	274	274	134	83.3
BFGS	Unconstrained	144	183	N/A	41.1	17.5
	Tridiagonal	215	276	N/A	60.8	24.6
	SPF	252	343	N/A	77.4	29.5
	CCF	399	536	N/A	117.5	44
Conjugate	Unconstrained	490	646	N/A	141.5	53.4
Gradient	Tridiagonal	546	716	N/A	155.5	59.2
	SPF	1165	1604	N/A	358.8	131.2
	CCF	5435	7755	N/A	1706.7	602.2
Steepest	Unconstrained	* 6037	8037	N/A	1743.8	616
Descent	Tridiagonal	* 6075	8377	N/A	1831.9	641.5
	SPF	* 6031	8031	N/A	1731.4	610.5
	CCF	* 5997	8498	N/A	1841.9	653.5

Table 2.1: Four Disk Example

of each other are not exploited to reduce computational effort. These numerical examples have been run on a 120 MHz, Pentium PC.

2.4.2 Observations

A sample of the results obtained are shown in Tables 2.1 and 2.2. The Quasi-Newton algorithm is more efficient than the continuation method for most cases. The continuation method is in general more efficient than the conjugate gradient method. The conjugate gradient method, as expected, is more efficient than the steepest descent method. The better performance of the Quasi-Newton algorithm over continuation, becomes more apparent as the dimension of the problem increases.

Example (Controller Order)	Basis	Function	Gradient	Hessian	Mflops	sec
Fourdisk (8th order)	Unconstrained	75	75	75	1261.3	220.1
	Tridiagonal	63	63	63	255.4	71.73
Fourdisk (6th order)	Unconstrained	72	72	72	350.1	94.3
	Tridiagonal	72	72	72	145.8	55.4
Fourdisk (4th order)	Unconstrained	58	58	58	68.8	36.7
	Tridiagonal	60	60	60	51	31.6
Navion (7th order)	Unconstrained	692	692	692	11938	2007.6
	Tridiagonal	592	592	592	3753.7	930.8
Navion (6th order)	Unconstrained	267	267	267	2537.3	523.2
	Tridiagonal	411	411	411	1746	506.4
Navion (4th order)	Unconstrained	108	108	108	234.4	94.9
	Tridiagonal	190	190	190	296.2	143.7
Spacecraft (6th order)	Unconstrained	101	101	101	428	119
	Tridiagonal	89	89	89	140.6	63.9
Spacecraft (4th order)	Unconstrained	52	52	52	46.7	31.7
	Tridiagonal	63	63	63	38.3	30.6

Table 2.2: Continuation: Unconstrained vs Tridiagonal

The * denotes failure to meet the normalized gradient tolerance within 1000 iterations. This occurs most often in the case of the steepest descent method due to oscillations close to the minimum, and is a well known deficiency of this method.

The numerical conditioning of the algorithms when using the tridiagonal basis was better than when using the second order polynomial form (SPF) and the controllability canonical form (CCF) and is apparently due to the fact that the tridiagonal form is a more general representation than SPF and CCF. In fact, SPF is a special case of the tridiagonal form.

For the continuation algorithms, the run times when the controller is constrained to the tridiagonal basis are considerably smaller than those for the unconstrained case. For the BFGS algorithm, the run times for the tridiagonal basis are slightly

larger than those for the unconstrained case. However, in both cases, as the controller dimension increases, the size of the parameter vector associated with the unconstrained "basis" increases much more rapidly than the parameter vector associated with the tridiagonal basis. Hence, the convergence times for the tridiagonal case increases much less rapidly than that for the unconstrained case, as the controller dimension increases. This effect is more pronounced in the case of the continuation method than the BFGS method.

2.5 Conclusions

In this chapter three examples have been used to compare the behavior of three standard descent algorithms with a recently developed continuation algorithm for H_2 optimal, reduced-order design. The results indicate that the Quasi-Newton (BFGS) algorithm is more efficient than the continuation algorithm which in turn is more efficient than the conjugate gradient and steepest descent algorithms. The second order polynomial form (SPF) and the controllability canonical form (CCF) are not very efficient bases and are subject to illconditioning problems. When using a tridiagonal basis (as opposed to the unconstrained "basis"), the advantage of a smaller parameter vector θ, starts to outweigh the disadvantage of reduced numerical conditioning due to a basis constraint, as the order of the controller is increased. Hence, the tridiagonal basis appears to be an excellent constraint basis for fixed-structure numerical algorithms.

CHAPTER 3

PROBABILITY-ONE HOMOTOPY ALGORITHMS FOR ROBUST CONTROLLER SYNTHESIS WITH FIXED-STRUCTURE MULTIPLIERS

3.1 Introduction

During the past two decades, major advancements have been made in robust control theory. Building upon H_∞ theory, the structured singular value (SSV) (Doyle 1982a, Packard and Doyle 1993) was defined as a nonconservative robustness measure for the analysis of linear systems with dynamic, arbitrary phase, multiple-block uncertainty. The supremum of the structured singular value over nonnegative frequencies is the inverse of the multivariable stability margin (see Safonov (1980), Safonov and Athans (1981) and the references therein). The initial developments in structured singular value theory focussed on dynamic uncertainty with arbitrary phase (often called "complex uncertainty") and hence, although less conservative than H_∞ theory, could still yield very conservative robustness bounds for systems with parametric uncertainty. This led to the development of mixed (i.e., real and complex) structured singular value (MSSV) theory (Fan *et al.* 1991, Young 1993) which considers block-diagonal uncertainty with both dynamic and real scalar parametric elements.

Parallel research addressed the issue of real parameter uncertainty using absolute stability theory such as Popov analysis (Haddad and Bernstein 1991, 1993, 1995a, Haddad *et al.* 1992, 1994c, 1996) and was developed by recognizing the relationship between sector bounded nonlinearities and interval bounds on linear uncertainties. This work was soon seen to provide an upper bound for the MSSV (Haddad *et al.* 1992, 1994c, How and Hall 1993). In fact, in contrast to the initial work on the

17

MSSV, this research provided the first *fixed-structure* multiplier versions of MSSV theory. A unique contribution of some of this work is that it led to the development of upper bounds on an H_2 cost functional over the uncertainty set under consideration. By optimizing this upper bound and using a Riccati equation constraint, continuation algorithms have been developed for MSSV controller synthesis (How *et al.* 1994a, 1994b, 1996). A related algorithm for complex structured singular value (CSSV) controller synthesis is given in Haddad *et al.* (1994a). Note that the H_2 approach allows the direct design of fixed-architecture (e.g., reduced-order or decentralized) controllers and the simultaneous optimization of the controller and (fixed-structure) multipliers, hence avoiding M-K (i.e., multiplier-controller) iteration schemes. However, to date the synthesis algorithms have been formulated only for the case of the the Popov multiplier. In addition, the algorithms rely on an *ad hoc* initialization scheme, have not used the prediction capabilities obtained by computing the Jacobian matrix of the homotopy (or continuation) map, and have assumed that the homotopy curve is monotonic.

A similar line of research has been developed independently in Chiang and Safonov (1992), Ly *et al.* (1994), Safonov and Chiang (1993). This work also provides a fixed-structure multiplier version of the MSSV but, unlike the approach described in Haddad and Bernstein (1991, 1993), Haddad *et al.* (1992, 1994c, 1996) this approach develops multipliers for strictly linear uncertainties. The associated robustness analysis was originally formulated in terms of a convex, frequency-domain optimization problem but has recently been reformulated in terms of a (convex) linear-matrix-inequality (LMI) problem (Ly *et al.* 1994, Balakrishnan *et al.* 1994). These results have led to the recognition that robust control design can be approached via solving a (nonconvex) "bilinear matrix inequality" (BMI) (Goh *et al.* 1994a, 1994b, Safonov *et al.* 1994). This approach allows the design of fixed-architecture controllers and can be implemented without using M-K iteration. To obtain a reasonably sized BMI, the multiplier set must be restricted to lie in the span of a stable basis (Goh *et al.* 1994a). However, the choice of this basis is unclear and can potentially introduce a high degree of conservatism. If the less conservative LMI formulation, requiring the use of unstable multipliers, is used, the resultant BMI is of very high dimension due to the introduction of a Lyapunov inequality of the dimension of the closed-loop sys-

tem to ensure closed-loop stability (Safonov *et al.* 1994). In contrast, the robustness analysis results using a Riccati equation formulation easily extend to robust control design without placing any basis restrictions on the multipliers or introducing high dimensionality.

In this chapter a Riccati equation constraint is used to formulate fixed-architecture, robust control design methods that use general forms of the fixed-structure multipliers. The proposed method relies on the development of an artificial cost function. This cost function also includes barrier functions to enforce positive definite constraints (e.g., on the Riccati solution P) which allows the constrained optimization problem (the constraints including $P > 0$) to be converted into an unconstrained optimization problem. The cost function is not physically meaningful so we do not encounter the normal problems associated with making the barrier functions small at the last step of the optimization process. (See Fletcher (1987) for a discussion of this negative feature of standard barrier function methods.) If the barrier terms are ignored and a certain term is added to the cost function, the cost function becomes an H_2 upper bound.

Due to the positive definite constraint on the Riccati solution, it is not possible to approach the solution to the optimization problem using standard descent methods. Hence, we develop probability-one homotopy algorithms (Watson 1987a, Watson *et al.* 1987b) to find the solution. This class of homotopy algorithms is distinct from classical continuation algorithms (Allgower and Georg 1990) in that they follow the zero curve using the arc length parameter and not the actual homotopy parameter λ. This allows the zero curve to be nonmonotonic in λ and provides additional numerical robustness. In addition, the algorithms developed here can be initialized with any stabilizing compensator and admissible multiplier, in contrast to the algorithms of How *et al.* (1994a, 1994b, 1996), and Haddad *et al.* (1994a).

3.2   Problem Formulation

Consider the standard uncertainty feedback configuration of Figure 3.1, where $G(s) \in C^{m \times m}$ is asymptotically stable and $G(s) \sim \left[\begin{array}{c|c} A & B \\ \hline C & D \end{array} \right]$. It is assumed that the
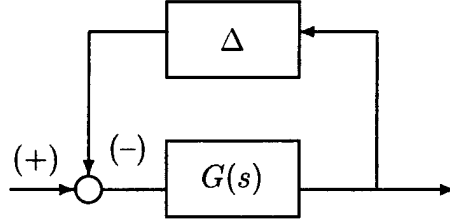
Figure 3.1: Standard Uncertainty Feedback Configuration

uncertainty $\Delta \in \mathcal{C}^{m \times m}$ belongs to the set

$$\boldsymbol{\Delta}_\gamma \triangleq \{\Delta = \text{block-diag}(\Delta_1, ..., \Delta_p) : \Delta_i \in \mathcal{I}_i, \ \sigma_{\max}(\Delta_i) \leq \gamma, i = 1, ..., p, \ \sum_{i=1}^{p} k_i = m\},$$
$$(3.1)$$

where $\mathcal{I}_i \subseteq \mathcal{C}^{k_i \times k_i}$ denotes the internal structure of the uncertainty block Δ_i and $\gamma > 0$.

We need to find sufficient conditions such that the negative feedback interconnection of $G(s)$ and Δ is asymptotically stable (or, equivalently, $\det(I + G(j\omega)\Delta) \neq 0$, $\omega \in \mathcal{R}$) for all $\Delta \in \boldsymbol{\Delta}_\gamma$. The sufficient conditions for robust stability (and performance) have been formulated as a Riccati equation feasibility problem and continuation algorithms have been developed to solve these problems. Below, we briefly present some of the most significant contributions of this research.

Riccati Equation Feasibility Problem (REFP).

Theorem 3.1. If there exists $\theta \in \mathcal{R}^q$, $\epsilon > 0$, and $P \in \mathcal{R}^{r \times r}$ such that

$$0 = \tilde{A}_\gamma^T(\theta)P + P\tilde{A}_\gamma(\theta) + (\tilde{B}_\gamma^T(\theta)P - \tilde{C}_\gamma(\theta))^T (\tilde{D}_\gamma(\theta) + \tilde{D}_\gamma^T(\theta))^{-1} (\tilde{B}^T(\theta)P - \tilde{C}_\gamma(\theta)) + \epsilon I,$$
$$(3.2)$$

$$P > 0, \quad \tilde{D}_\gamma(\theta) + \tilde{D}_\gamma^T(\theta) > 0, \quad (3.3)$$

where θ corresponds to the free parameters of the matrices providing a state-space representation of the compatible multiplier and $\tilde{A}_\gamma$, $\tilde{B}_\gamma$, $\tilde{C}_\gamma$, $\tilde{D}_\gamma$ are functions of the plant and multiplier matrices, then the negative feedback interconnection of $G(s)$ and Δ is asymptotically stable.

The dimension q is determined by the multiplier and r is determined by both the multiplier and the nominal plant size.

If we are considering control design then θ corresponds to the free parameters of both multiplier and controller matrices. The controller matrices essentially provide extra degrees of freedom to satisfy the Riccati equation constraint (3.2). Note that $\tilde{A}_\gamma(\theta)$, $\tilde{B}_\gamma(\theta)$, $\tilde{C}_\gamma(\theta)$, and $\tilde{D}_\gamma(\theta)$ are generally nonlinear functions of θ. Hence it is not possible to convert the REFP to an LMI feasibility problem.

We approach the development of a solution technique by defining the following artificial cost function

$$J(\theta, \epsilon, P) = \alpha \theta^T \theta + \alpha \epsilon^2 + r_d \, \text{tr} \left[\tilde{D}_\gamma(\theta) + \tilde{D}_\gamma^T(\theta) \right]^{-1} + r_p \, \text{tr} \, P^{-1} + r_\epsilon \frac{1}{\epsilon}, \qquad (3.4)$$

where α, r_d, r_p, and r_ϵ are positive scalars.

To characterize the extremals, we define the Lagrangian

$$\mathcal{L}(\theta, \epsilon, P, Q) = J(\theta, \epsilon, P) + \text{tr} \, QW(\theta, \epsilon, P), \qquad (3.5)$$

where $W(\theta, \epsilon, P)$ denotes the right hand side of (3.2) and Q is the symmetric Lagrange multiplier. Note that the constraints (3.3) are absorbed into J as barriers. The necessary conditions are given by Fletcher (1987)

$$0 = \frac{\partial \mathcal{L}}{\partial \theta}, \quad 0 = \frac{\partial \mathcal{L}}{\partial \epsilon}, \qquad (3.6)$$

$$
\begin{aligned}
0 = \frac{\partial \mathcal{L}}{\partial Q} &= \left(\tilde{B}_\gamma^T(\theta)P - \tilde{C}_\gamma(\theta) \right)^T \left(\tilde{D}_\gamma(\theta) + \tilde{D}_\gamma^T(\theta) \right)^{-1} \left(\tilde{B}^T(\theta)P - \tilde{C}_\gamma(\theta) \right) \\
&\quad + \tilde{A}_\gamma^T(\theta)P + P\tilde{A}_\gamma(\theta) + \epsilon I,
\end{aligned}
\qquad (3.7)
$$

$$0 = \frac{\partial \mathcal{L}}{\partial P} = \tilde{F}_\gamma Q + Q\tilde{F}_\gamma^T - r_p(P^{-2}) + \frac{\beta}{\epsilon} V^T, \qquad (3.8)$$

where

$$\tilde{F}_\gamma = \tilde{A}_\gamma - \tilde{B}_\gamma \left[\tilde{D}_\gamma + \tilde{D}_\gamma^T \right]^{-1} \left[\tilde{B}_\gamma P - \tilde{C}_\gamma \right].$$

Although (3.6)-(3.8) characterizes extremals, we have not yet developed a reliable method to compute these extremals. Note that standard interior point descent methods (e.g., steepest descent, conjugate gradient, or quasi-Newton methods) cannot be directly applied due to the nature of the constraints. For example, suppose we attempt to initialize one of these methods with a multiplier (in the class of multipliers for the given uncertainty set) represented by θ_0 and also choose an initial ϵ denoted

by ϵ_0. Then, if there exists a positive-definite solution P_0 to (3.2), the REFP is solved and there is no need for further computations. However, suppose there is no positive definite solution P_0 to (3.2). Then, $(\theta_0, \epsilon_0, P_0)$ cannot be used to initialize an interior point descent method to find a solution to the optimization problem since this class of methods requires an initial feasible interior point. What is needed is a numerical technique that is able to find a solution by starting with a nonfeasible point $(\theta_0, \epsilon_0, P_0)$. This is accomplished in the next section using a probability-one homotopy algorithm.

3.3 Probability-One Homotopy Algorithms for Robust Controller Synthesis

Consider a function $F : \mathcal{R}^N \times \mathcal{R} \to \mathcal{R}^N$ that is $\mathcal{C}^2$. Given $\gamma_f \in \mathcal{R}$, it is desired to find $x \in \mathcal{R}^N$ such that

$$0 = F(x, \gamma_f). \tag{3.9}$$

This is a standard zero finding problem. In the context of the robustness analysis results of the previous section

$$x = (\theta, \epsilon), \quad N = q + 1, \tag{3.10}$$

$$F(x, \gamma) = \left(\frac{\partial \mathcal{L}}{\partial \theta}, \frac{\partial \mathcal{L}}{\partial \epsilon} \right), \tag{3.11}$$

and γ_f corresponds to some desired lower bound on the multivariable stability margin. Note that $0 = \frac{\partial \mathcal{L}}{\partial Q}$ and $0 = \frac{\partial \mathcal{L}}{\partial P}$ are implicitly satisfied by choosing P as the (maximal) solution of the Riccati equation (3.7) (or (3.2)) and Q as the solution of the Lyapunov equation (3.8).

Let $x_0 = [\theta_0, \ \epsilon_0]$ represent a feasible multiplier, a stabilizing compensator and a positive ϵ. Furthermore let γ_0 be chosen small enough such that there exists a positive-definite solution P_0 to (3.2). (It is assumed that $\gamma_0 < \gamma_f$ such that the robustness problem is not trivial.)

We let

$$\gamma(\lambda) = (1 - \lambda)\gamma_0 + \lambda\gamma_f \tag{3.12}$$

and define the probability-one homotopy map $\rho : [0, 1) \times \mathcal{R}^N \to \mathcal{R}^N$ by

$$\rho(\lambda, x) = \lambda F(x, \gamma(\lambda)) + (1 - \lambda)(x - x_0). \tag{3.13}$$

Obviously, $0 = \rho$ has the unique solution x_0 and $\rho = F(x, \gamma_f)$. These are necessary conditions for the homotopy map. In the context of the robustness problem, this homotopy map has the desirable property that it can be initialized with any feasible multiplier. In addition, for any $\lambda \in [0, 1)$ the corresponding point on the zero curve (x, λ) corresponds to a controller and multiplier that guarantees the level of robustness corresponding to $\gamma(\lambda)$ since the Riccati equation (3.2) (or (3.7)) with the constraints (3.3) are satisfied with $\gamma = \gamma(\lambda)$. Hence, each point on the zero curve $(0 = \rho(\lambda, x), \ \lambda \in [0, 1))$, is physically meaningful even though $F(x, \gamma(\lambda)) \neq 0$ for $0 < \lambda < 1$.

3.3.1 Probability-One Homotopy Algorithm

Complete details of the numerical algorithm are in Watson *et al.* (1987b). A sketch is as follows.

1. Set $\lambda \triangleq 0$, $x \triangleq x_0$.

2. Evaluate the homotopy map ρ and the Jacobian of the homotopy map $D\rho$.

3. Predict the next point $Z^{(0)}$ on the homotopy zero curve using, e.g., a Hermite cubic interpolant.

4. For $k = 0, 1, 2, \ldots$ until convergence do

$$Z^{(k+1)} = Z^{(k)} - [D\rho(Z^{(k)})]^\dagger \rho(Z^{(k)}),$$

where $[D\rho(Z)]^\dagger$ is the Moore-Penrose pseudo inverse of $D\rho(Z)$. Let $(x_1, \lambda_1) = \lim_{k \to \infty} Z^k$.

5. If $\lambda_1 < 1$, then set $x = x_1$, $\lambda = \lambda_1$, and go to to step (2).

6. If $\lambda_1 > 1$, compute the solution x at $\lambda = 1$ using, e.g., inverse linear interpolation (Watson *et al.* 1987b).

3.3.2 Robust Control Synthesis Using the Popov Multiplier for a Benchmark Problem

To illustrate robust control synthesis with the probability-one homotopy algorithm, we consider the two-mass/spring benchmark system shown in Figure 3.2 with

uncertain stiffness k. A control force acts on the body 1 and the position of body 2 is measured, resulting in a noncollocated control problem. This benchmark problem is discussed in detail in Wei and Bernstein (1992).
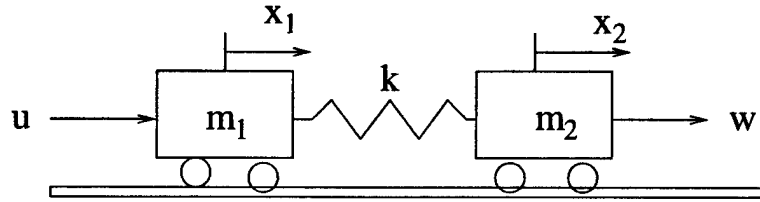

Figure 3.2: Two Mass/Spring System

We desire to design a constant gain linear feedback compensator of the form

$$\dot{x}_c(t) = A_c x_c(t) + B_c y(t), \tag{3.14}$$

$$u(t) = -C_c x_c(t), \tag{3.15}$$

such that the closed-loop system is stable for $0.5 < k < 2.0$ and for a unit impulse disturbance at $t = 0$, the performance variable z has a settling time of about 15 s for the nominal system (with $k = k_{nom} = 1$).

The controller transfer function obtained by the probability-one homotopy algorithm and the Popov multiplier $H^2 + Ns$, is given by

$$H(s) = \frac{2819\,(s + 0.2079)[(s - 0.0978)^2 + 0.8063^2]}{[(s + 4.004)^2 + 1.8294^2][(s + 3.4747)^2 + 9.9745^2]}. \tag{3.16}$$

This controller is guaranteed by the theory to be robust for the range $0.5 < k < 2.0$ and this was also verified by a direct search. The settling time for the system was chosen to be the time required for the displacement of mass 2 to reach and stay within the interval [-0.1m, 0.1m]. The controller is seen to satisfy the settling time objective when connected to the nominal model corresponding to $k = 1$ N/m, as can be seen from the impulse response of the closed-loop system in Figure 3.3. It can also be seen that the settling time objective is satisfied for the entire family of plants $(0.5 < k < 2.0)$, which, though not a design requirement, is a very desirable characteristic of the controller.
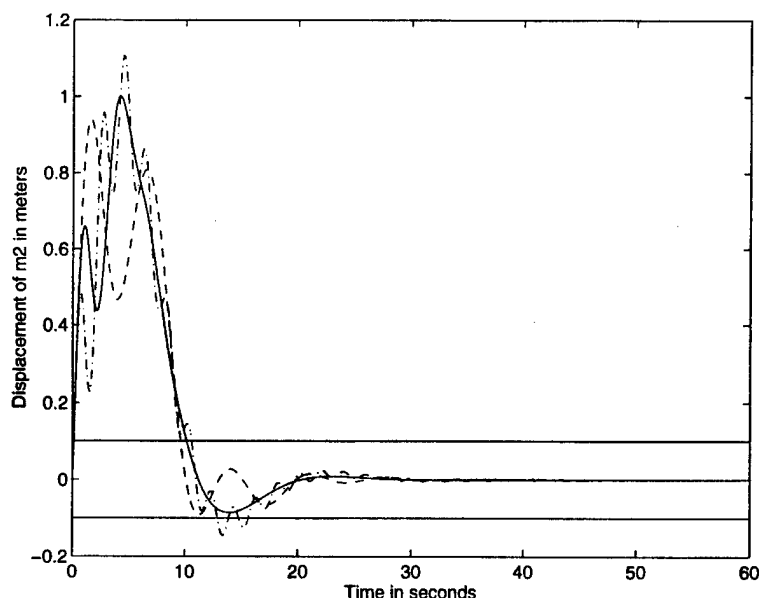
Figure 3.3: Impulse Response of Closed-Loop System

3.4 Conclusion

It has been demonstrated that fixed-architecture, robust control design using general fixed-structure multipliers can be formulated as a Riccati equation feasibility problem, a nonlinear, algebraic feasibility problem. Probability-one homotopy algorithms have been proposed to solve this feasibility problem. These algorithms differ from previously developed continuation algorithms, developed exclusively for the case of the Popov multiplier, in that they can be initialized with any admissible multiplier and stabilizing compensator. Like other probability-one homotopy algorithms they also tend to be better conditioned than the alternative continuation algorithms. The results have been specialized to the special case of Popov multipliers and the use of the algorithm has been illustrated by implementing it for the synthesis of fixed-structure controllers with robust H_2 performance for a standard benchmark problem.

CHAPTER 4

SYNTHESIS OF FIXED-ARCHITECTURE, ROBUST H_2 AND H_∞ CONTROLLERS

4.1 Introduction

This chapter considers the design of robust controllers using the state space Popov analysis criterion which is based on the Popov stability multiplier $W(s) = H^2 + Ns$. This is a special case of mixed structured singular value synthesis (Haddad *et al.* 1994c, How *et al.* 1993). Algorithms for both robust H_2 and H_∞ performance are described and compared. The formulations which closely follow those presented in Collins *et al.* (1996c, 1997a) require the minimization of a cost functional subject to a Riccati equation constraint. These formulations have several advantages. First, compensator order and architecture can be specified a priori. In addition, both the controller and multiplier parameters can be optimized simultaneously which avoids M-K (i.e., multiplier-controller) iteration, potentially leading to better performing robust controllers. For robust H_2 performance the cost function that is minimized is an upper bound on the H_2 performance over the uncertainty set. For H_∞ performance, an artificial cost function is used.

Because of positive definite constraints on the Riccati equation solution, standard descent techniques cannot be used to solve the resulting optimization problem. Hence, probability-one homotopy algorithms have been formulated (Collins *et al.* 1996c, 1997a). These algorithms have desirable properties when applied to controller design. First, they can be initialized with any feasible multiplier and stabilizing controller. Also, each controller computed as the homotopy curve is traversed is physically meaningful. In particular, for the robust H_2 performance each controller along the homotopy path guarantees a specified degree of robust stability while for

the robust H_∞ performance problem each controller guarantees a specified degree of both robust stability and robust performance. Collins *et al.* (1996c, 1997a) describe implementation of the algorithm for H_2 performance. A major contribution described in this chapter is the implementation of the algorithm for robust H_∞ performance and a comparison with the algorithm for robust H_2 performance.

4.2 Riccati Equation Approaches to Robust Controller Synthesis Using the Popov Multiplier

Consider the uncertainty feedback system shown in Fig. 4.1, where $G(s)$ has the nth order, stabilizable and detectable realization

$$
G(s) \sim \left[\begin{array}{c|ccc} A & B_0 & D_1 & B \\ \hline C_0 & 0 & 0 & 0 \\ E_1 & 0 & 0 & 0 \\ C & 0 & D_2 & 0 \end{array} \right],
\tag{4.1}
$$

$K(s)$ has a realization of order $n_c \leq n$ given by

$$
K(s) \sim \left[\begin{array}{c|c} A_c & B_c \\ \hline -C_c & 0 \end{array} \right],
\tag{4.2}
$$

and $\Delta_u \in \mathcal{U}$ where for M_1 and M_2 in $\mathcal{D}^{m_0 \times m_0}$ with $M_2 - M_1 > 0$, $\mathcal{U}$ is the real parametric uncertainty set

$$
\mathcal{U} \triangleq \{\Delta_u \in \mathcal{R}^{m_0 \times m_0} : M_1 < \Delta_u < M_2\}.
\tag{4.3}
$$

Let

$$
\tilde{z} = [z \quad E_2 u]^T
\tag{4.4}
$$

and let θ be a vector representation of the controller state space matrices, for example

$$
\theta = \left[\begin{array}{ccc} \text{vec}^T(A_c) & \text{vec}^T(B_c) & \text{vec}^T(C_c) \end{array} \right]^T.
\tag{4.5}
$$

Then Fig. 4.1 is equivalent to Fig. 4.2 where

$$
\tilde{G}(s, K) \sim \left[\begin{array}{c|cc} \tilde{A}(\theta) & \tilde{B}_0 & \tilde{D}(\theta) \\ \hline \tilde{C}_0 & 0 & 0 \\ \tilde{E}(\theta) & 0 & 0 \end{array} \right]
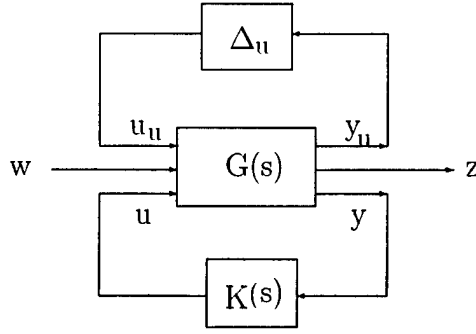\tag{4.6}
$$

Figure 4.1: Uncertain Feedback System

where

$$\tilde{A}(\theta) = \begin{bmatrix} A & -BC_c \\ B_cC & A_c \end{bmatrix}, \tag{4.7}$$

$$\tilde{B}_0 = \begin{bmatrix} B_0 \\ 0 \end{bmatrix}, \quad \tilde{D}(\theta) = \begin{bmatrix} D_1 \\ B_cD_2 \end{bmatrix}, \tag{4.8}$$

$$\tilde{C}_0 = \begin{bmatrix} C_0 & 0 \end{bmatrix}, \quad \tilde{E}(\theta) = \begin{bmatrix} E_1 & 0 \\ 0 & E_2C_c \end{bmatrix}. \tag{4.9}$$

It is desired to determine $K(s)$ or equivalently θ such that for all $\Delta_u \in \mathcal{U}$, the system of Fig. 4.2 is asymptotically stable and either $\|\mathcal{F}_u(\tilde{G}, \Delta_u)\|_2$ or $\|\mathcal{F}_u(\tilde{G}, \Delta_u)\|_\infty$ satisfies some prespecified bounds.

Define

$$\tilde{R}(\theta) \triangleq \tilde{E}^T(\theta)\tilde{E}(\theta), \quad \tilde{V}(\theta) \triangleq \tilde{D}(\theta)\tilde{D}^T(\theta). \tag{4.10}$$

The next theorem formulates a synthesis problem for robust H_2 performance in terms of the Popov multiplier $W(s) = H^2 + Ns$.

Theorem 4.1. Suppose $\tilde{G}(s, K)$ is asymptotically stable. If there exist θ, $H \in \mathcal{D}^{m_0 \times m_0}$, $N \in \mathcal{D}^{m_0 \times m_0}$, $P > 0$, and $\epsilon > 0$ such that

$$Y = \begin{bmatrix} 2H^2(M_2 - M_1)^{-1} + N\tilde{C}_0\tilde{B}_0 + \tilde{B}_0^T\tilde{C}_0^T N \end{bmatrix} > 0 \tag{4.11}$$

and

$$\begin{aligned} 0 = \ & (\tilde{A}(\theta) - \tilde{B}_0M_1\tilde{C}_0)^T P + P(\tilde{A}(\theta) - \tilde{B}_0M_1\tilde{C}_0) + \\ & [\tilde{B}_0^T P - H^2\tilde{C}_0 - N\tilde{C}_0(\tilde{A}(\theta) - \tilde{B}_0M_1\tilde{C}_0)]^T \cdot Y^{-1} \cdot \end{aligned}$$

$$[\tilde{B}_0^T P - H^2 \tilde{C}_0 - N\tilde{C}_0(\tilde{A}(\theta) - \tilde{B}_0 M_1 \tilde{C}_0)] +$$
$$\epsilon \tilde{R}(\theta), \tag{4.12}$$

then the uncertain system of Fig. 4.2 is asymptotically stable for each $\Delta_u \in \mathcal{U}$. In addition,

$$\max_{\Delta_u \in \mathcal{U}} \|\mathcal{F}_u(\tilde{G}, \Delta_u)\|_2 \leq J(\epsilon, \theta, H, N, P) \tag{4.13}$$

$$\triangleq \frac{1}{\epsilon} \mathrm{tr}[P + \tilde{C}_0^T(M_2 - M_1)N\tilde{C}_0]\tilde{V}(\theta).$$

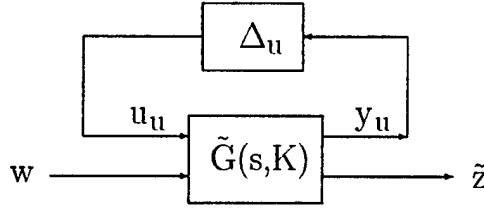**Proof** See Haddad *et al.* (1994c). □

Figure 4.2: Closed-Loop Representation of Uncertain System

To consider H_∞ performance, a fictitious complex uncertainty block Δ_p is inserted into Fig. 4.2 (Doyle *et al.* 1982b, Packard and Doyle 1993) as shown in Fig. 4.3. It is assumed that $\sigma_{max}(\Delta_p) < \gamma$. For ease of presentation assume that $\dim(\tilde{z}) = \dim(w) = q$, such that $\Delta_p \in \mathcal{C}^{q \times q}$. Define

$$\bar{M}_1 \triangleq \text{block-diag}\{M_1, -\gamma I_q\}, \quad \bar{M}_2 \triangleq \text{block-diag}\{M_2, \gamma I_q\} \tag{4.14}$$

$$\tilde{B}(\theta) \triangleq \left[\begin{array}{cc} \tilde{B}_0 & \tilde{D}(\theta) \end{array}\right], \quad \tilde{C}(\theta) \triangleq \left[\begin{array}{c} \tilde{C}_0 \\ \tilde{E}(\theta) \end{array}\right]. \tag{4.15}$$

The next theorem formulates a synthesis problem for robust H_∞ performance in terms of the Popov multiplier $W(s) = H^2 + Ns$.

Theorem 4.2. Suppose $\tilde{G}(s, K)$ is asymptotically stable. If there exist θ, $H = \text{block-diag}\{H_1, H_2\}$ where $H_1 \in \mathcal{D}^{m_0 \times m_0}$ and $H_2 \in \mathcal{R}^{q \times q}$ satisfies $H_2 \Delta_p = \Delta_p H_2$, $N = \text{block-diag}\{N_1, 0_q\}$ where $N_1 \in \mathcal{D}^{m_0 \times m_0}$, $P > 0$ and $\epsilon > 0$ such that

$$\tilde{Y} = \left[\begin{array}{c} 2H^2(\bar{M}_2 - \bar{M}_1)^{-1} + N\tilde{C}\tilde{B} + \tilde{B}^T\tilde{C}^T N \end{array}\right] > 0 \tag{4.16}$$

and

$$0 = (\tilde{A}(\theta) - \tilde{B}M_1\tilde{C})^T P + P(\tilde{A}(\theta) - \tilde{B}\bar{M}_1\tilde{C}) +$$
$$[\tilde{B}^T P - H^2\tilde{C} - N\tilde{C}(\tilde{A}(\theta) - \tilde{B}\bar{M}_1\tilde{C})]^T \cdot \tilde{Y}^{-1} \cdot$$
$$[\tilde{B}^T P - H^2\tilde{C} - N\tilde{C}(\tilde{A}(\theta) - \tilde{B}\bar{M}_1\tilde{C})] + \epsilon I, \qquad (4.17)$$

then the uncertain system of Fig. 4.3 is asymptotically stable for each $\Delta_u \in \mathcal{U}$. In addition,

$$\max_{\Delta_u \in \mathcal{U}} \|\mathcal{F}_u(\tilde{G}, \Delta_u)\|_\infty < \frac{1}{\gamma}. \qquad (4.18)$$

Proof. Follows from results in Haddad *et al.* (1994c) and Haddad *et al.* (1995b, 1996) and a straight forward variant of the main loop theorem (Packard and Doyle, 1993). □
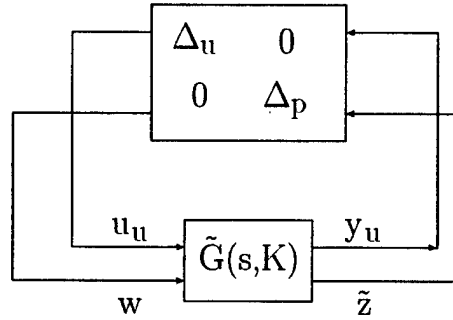
Figure 4.3: Closed-Loop Uncertain System with 'Performance Block'

4.3 Algorithms for Robust Controller Synthesis

Theorems 4.1 and 4.2 both pose robust controller synthesis as a Riccati Equation Feasibility Problem (REFP) (Collins *et al.* 1996c, 1997a). As discussed in Collins *et al.* (1996c, 1997a) an approach to solving the REFP of either Theorem 4.1 or Theorem 4.2 can be based on solving an optimization problem

$$\min_{\epsilon, \theta, H, N, P} J(\epsilon, \theta, H, N, P) \text{ subject to (4.12) or (4.17)} \qquad (4.19)$$

where $J(\cdot)$ denotes an appropriate cost functional.

For control design for robust H_2 performance $J(\cdot)$ is given by (4.13). For robust H_∞ performance $J(\cdot)$ can be chosen to minimize the artificial cost function

$$J(\epsilon, \theta, H, N, P) = \text{tr} P. \tag{4.20}$$

To characterize the extremals define the Lagrangian

$$\mathcal{L}(\epsilon, \theta, H, N, P, Q) = J(\epsilon, \theta, H, N, P) + \text{tr} Q W(\epsilon, \theta, H, N, P) \tag{4.21}$$

where $W(\cdot)$ denotes the right hand side of (4.12) or (4.17). The necessary conditions for a solution to (4.19) are given by

$$0 = \frac{\partial \mathcal{L}}{\partial \epsilon}, \quad 0 = \frac{\partial \mathcal{L}}{\partial \theta}, \quad 0 = \frac{\partial \mathcal{L}}{\partial H}, \quad 0 = \frac{\partial \mathcal{L}}{\partial N}, \tag{4.22}$$

$$0 = \frac{\partial \mathcal{L}}{\partial Q}, \quad 0 = \frac{\partial \mathcal{L}}{\partial P}. \tag{4.23}$$

In this chapter probability-one homotopy algorithms based on the Popov multiplier have been developed and implemented for both robust H_2 and robust H_∞ controllers. The controllers and control algorithms are then compared with each other and with that produced using complex singular value synthesis.

4.4 Numerical Example

To illustrate robust control synthesis with the probability-one homotopy algorithm, we consider the two-mass/spring benchmark system shown in Figure 3.2 with uncertain stiffness k. A control force acts on the body 1 and the position of body 2 is measured, resulting in a noncollocated control problem. This benchmark problem is discussed in detail in Wei and Bernstein (1992).

We desire to design a constant gain linear feedback compensator $K(s)$ with realization (4.2) such that the closed-loop system is stable for $0.5 < k < 2.0$ and for a unit impulse disturbance at $t = 0$, the performance variable z has a settling time of about 15 s for the nominal system (with $k = k_{\text{nom}} = 1$).

Observations

All three controllers are guaranteed by the theory to be robust for the range $0.5 <$
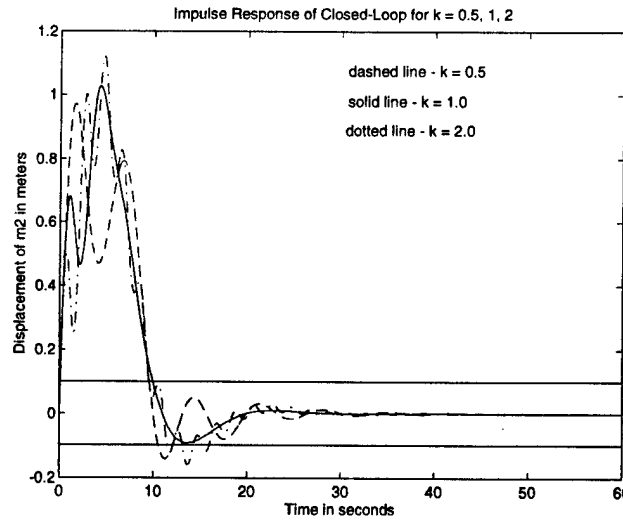
Figure 4.4: Impulse Response - Robust H_2 Controller

$k < 2.0$ and this was also verified by a direct search. It is seen that the upper bound on the worst case cost for both the robust H_2 and robust H_∞ controllers are fairly 'tight', whereas that for complex μ synthesis is clearly very conservative. The robust H_2 controller is stable for $0.35 < k < 2.39$; the robust H_∞ controller is stable for $0.4 < k < 2.45$ and the controller obtained by complex μ synthesis is stable for $0.32 < k < 6.7$. Clearly the controllers obtained using the Popov multiplier approach are less conservative than that obtained by complex μ synthesis.

The settling time for the system was chosen to be the time required for the displacement of mass 2 to reach and stay within the interval [-0.1m, 0.1m]. All three controllers are seen to satisfy the settling time objectives when connected to the nominal model corresponding to $k = 1$ N/m. It is seen that the settling time objective is satisfied for the entire family of plants ($0.5 < k < 2.0$), which, though not a design requirement, is a very desirable characteristic of the controllers. It is seen that the robust H_2 and robust H_∞ controllers obtained using the Popov multiplier approach yield similar time responses. It is seen that nearly similar control effort is required by both the robust H_2 and the robust H_∞ controllers and it is significantly less than that required by the complex μ controller. It is also seen from Fig. 4.5 that both the robust H_2 and the robust H_∞ controllers have bandwidths which are significantly smaller than the bandwidth of the complex μ controller.
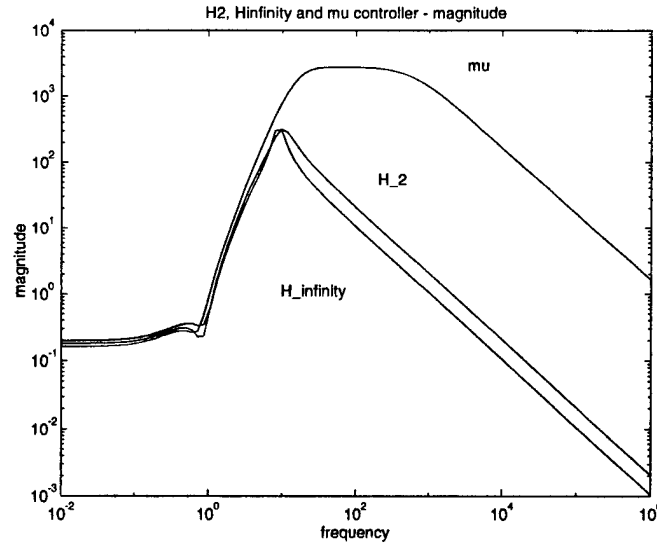
Figure 4.5: Controller Bandwidths

It is observed that the algorithm for robust H_∞ performance is much more computationally intensive than that for robust H_2 performance. This is because the expressions for the gradient and Hessian for H_∞ design are far more complex than those for H_2 design.

4.5 Conclusions

The Popov Multiplier has been used to develop probability-one homotopy algorithms for the design of robust controllers with guaranteed H_2 or H_∞ performance. The formulation closely follows that presented in Collins *et al.* (1996c, 1997a) and extends it to the case of robust controllers with H_∞ performance. Though the formulation for both the robust H_2 and the robust H_∞ problems are very similar, the gradient and the Hessian expressions for the H_∞ formulation are more complex. A numerical benchmark example is presented for both the robust H_2 and H_∞ controllers. Both controllers are found to have smaller bandwidth, smaller control authority and to be significantly less conservative than controllers obtained by complex μ synthesis. It is seen that the algorithms for the robust H_∞ controllers are more computationally intensive than algorithms for robust H_2 controllers, as is expected.

Certainly if the uncertainty is mixed, and the performance requirements are in terms of H_∞ cost, it is preferable to use the multiplier based algorithms with guaran-

teed H_∞ performance (as described in this chapter) than complex μ synthesis. The fact that the robust H_2 and the robust H_∞ algorithms produce controllers with similar characteristics, suggests that when the performance specifications are not directly in terms of either H_2 or H_∞ cost, one may use either of the two algorithms. In this case, due to the significant difference in computational complexity, it is advantageous to use the algorithm for H_2 performance.

CHAPTER 5

ROBUST CONTROLLER SYNTHESIS VIA NONLINEAR MATRIX INEQUALITIES

5.1 Introduction

Mixed structured singular value (MSSV) theory (Fan *et al.* 1991) was developed to nonconservatively analyze the robust stability and performance of systems with both real parametric and complex uncertainty. The LMI formulations of MSSV theory (Balakrishnan *et al.* 1994, Ly *et al.* 1994) led to the recognition that robust control design can be approached via solving a (nonconvex) "bilinear matrix inequality" (BMI) (Goh *et al.* 1994a, Safonov *et al.* 1994). This approach, like those based on a Riccati equation constraint (Collins *et al.* 1997a, Haddad and Bernstein 1991), allows the design of fixed-architecture controllers and can be implemented without using M-K iteration. To obtain a reasonably sized BMI, the multiplier set must be restricted to lie in the span of a stable basis (Goh *et al.* 1994a). However, the choice of this basis is unclear and can potentially introduce a high degree of conservatism. If the less conservative LMI formulation, requiring the use of unstable multipliers, is used, to ensure closed-loop stability, the resultant BMI must be of very high dimension due to the introduction of a Lyapunov inequality of the dimension of the closed-loop system (Safonov *et al.* 1994).

In this chapter the LMI approach to MSSV analysis is used to develop an approach to robust controller synthesis that is based on the stable factors of the multipliers and does not require the multipliers to be restricted to a basis. It is shown that this approach requires the solution of nonlinear matrix inequalities (NMI's). A continuation algorithm is presented for the solution of NMI's. This algorithm provides an alternative to the approaches proposed in Goh *et. al.* (1994c) to solve BMIs. The
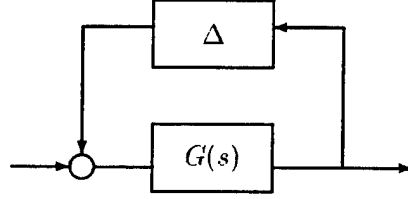
Figure 5.1: Standard Uncertainty Feedback Configuration

primary computational burden of the continuation algorithm is the solution of a series of LMIs. The Popov multiplier is used to formulate a NMI to solve a benchmark problem and the algorithm is used to synthesize a controller that meets the design constraints.

5.2 Multiplier Methods in Robustness Analysis

In this section we review the framework for mixed uncertainty robustness analysis with fixed-structure multipliers. The exposition generally follows that presented in Haddad *et al.* (1995b, 1996), Ly *et al.* (1994), Safonov and Chiang (1993), and Balakrishnan *et al.* (1994). We begin by considering the standard uncertainty feedback configuration of Figure 5.1, where $G(s) \in C^{m \times m}$ is asymptotically stable and $G(s) \sim \left[\begin{array}{c|c} A & B \\ \hline C & D \end{array} \right]$. It is assumed that the uncertainty $\Delta \in C^{m \times m}$ belongs to the set

$$\Delta_\gamma \triangleq \{\Delta = \text{block-diag}(\Delta_1, ..., \Delta_p) : \Delta_i \in \mathcal{I}_i, \ \sigma_{\max}(\Delta_i) \leq \gamma, i = 1, ..., p, \ \sum_{i=1}^{p} k_i = m\},$$

(5.1)

where $\mathcal{I}_i \subseteq C^{k_i \times k_i}$ denotes the internal structure of the uncertainty block Δ_i and $\gamma > 0$.

We need to find sufficient conditions such that the negative feedback interconnection of $G(s)$ and Δ is asymptotically stable (or, equivalently, $\det(I + G(j\omega)\Delta) \neq 0$, $\omega \in \mathcal{R}$) for all $\Delta \in \Delta_\gamma$. The sufficient conditions for robust stability (and performance) have been formulated as nonlinear matrix inequalities (NMIs). Continuation type algorithms have then been developed to solve these NMIs. Below, we briefly present some of the most significant contributions of this research.

Nonlinear Matrix Inequality for Robust Controller Synthesis (NMIRCS) .

Theorem 5.1. If there exists $\theta \in \mathcal{R}^q$, $\epsilon > 0$, and $P \in \mathcal{R}^{r \times r} > 0$ such that $\tilde{D}_\gamma + \tilde{D}_\gamma^T > 0$ and

$$\begin{bmatrix} \tilde{A}_\gamma^T(\theta)P + P\tilde{A}_\gamma(\theta) & -P\tilde{B}_\gamma(\theta) + \tilde{C}_\gamma^T(\theta) \\ -\tilde{B}_\gamma^T(\theta)P + \tilde{C}_\gamma(\theta) & (\tilde{D}_\gamma(\theta) - \epsilon I) + (\tilde{D}_\gamma(\theta) - \epsilon I)^T \end{bmatrix} > 0 \qquad (5.2)$$

where θ corresponds to the free parameters of the matrices providing a state-space representation of the compatible multiplier and $\tilde{A}_\gamma$, $\tilde{B}_\gamma$, $\tilde{C}_\gamma$, $\tilde{D}_\gamma$ are functions of the plant and multiplier matrices, then the negative feedback interconnection of $G(s)$ and Δ is asymptotically stable.

The dimension q is determined by the multiplier and r is determined by both the multiplier and the nominal plant size. If we are considering control design then θ corresponds to the free parameters of both multiplier and controller matrices. The controller matrices essentially provide extra degrees of freedom to satisfy the NMI constraint (5.2).

5.3 A Continuation Algorithm for NMI Feasibility Problems

Let $G : \mathcal{R}^n \times \mathcal{R} \to \mathcal{R}^{p \times p}$ be a nonlinear function and define

$$F_{\gamma^*}(x) \triangleq G(x, \gamma)|_{\gamma = \gamma^*}. \qquad (5.3)$$

Given γ_f, we desire to find x such that

$$F_{\gamma_f}(x) < 0. \qquad (5.4)$$

Let $\gamma : [0, 1) \to \mathcal{R}$ be the function

$$\gamma(\lambda) = (1 - \lambda)\gamma_0 + \lambda\gamma_f. \qquad (5.5)$$

It is assumed that given a value γ_0, there exists an easily computed point x_0 such that

$$F_{\gamma_0}(x_0) < 0. \qquad (5.6)$$

Define $H : \mathcal{R}^n \times [0, 1) \to \mathcal{R}^{p \times p}$ by

$$H(x, \lambda) \triangleq G(x, \gamma(\lambda)). \tag{5.7}$$

Consider the NMI

$$H(x, \lambda) < 0 \tag{5.8}$$

and note that

$$H(x, 0) = F_{\gamma_0}(x) \tag{5.9}$$

and hence at $\lambda = 0$, (5.8) has a solution $x = x_0$, which is easily found. Also,

$$H(x, 1) = F_{\gamma_f}(x), \tag{5.10}$$

and hence at $\lambda = 1$, (5.8) becomes the desired NMI (5.4).

To enable path following, instead of solving the NMI feasibility problem (5.4), we solve the following NMI eigenvalue problem

$$\min_x \lambda_{\max}(F_{\gamma_f}(x)) \tag{5.11}$$

where $\lambda_{\max}$ denotes the maximum eigenvalue of $F_{\gamma_f}(x)$. Clearly there exists a solution $x \in \mathcal{R}^n$ to the NMI feasibility problem (5.4) if and only if

$$\min_x \lambda_{\max}(F_{\gamma_f}(x)) < 0. \tag{5.12}$$

5.3.1 Continuation Algorithm

1. Set $\lambda \leftarrow 0$, $\gamma \leftarrow \gamma_0$, $x^{(0)} \leftarrow x_0$, $R_1 > 0$, $R_2 > 0$, $\Delta\lambda > 0$.

2. For $k = 0, 1, 2, \ldots$ until convergence compute $x^{(k+1)}$ by solving the LMI optimization problem

$$\min_{dx^{(k)}} \lambda_{\max}\left(A(dx^{(k)}, \lambda)\right)$$

subject to the LMI

$$A(dx^{(k)}, \lambda) < 0$$

and the move limit constraint

$$\left\| dx^{(k)} \right\| \leq R_1,$$

where

$$A(dx^{(k)}, \lambda) = H(x^{(k)}, \lambda) + \sum_i \frac{\partial H}{\partial x_i}\Big|_{x=x^{(k)}} dx_i^{(k)}$$

denotes the linearization of $H(x, \lambda)$ about $x^{(k)}$. $x^{(k+1)}$ is defined by

$$x^{(k+1)} = x^{(k)} + dx^{(k)}.$$

Let $x(\lambda) = \lim_{k\to\infty} x^{(k)}$.

3. If $\lambda = 1$, stop; $x(1)$ is the solution.

4. Compute the direction vector $z(\lambda)$ by solving the LMI optimization problem

$$\min_{z(\lambda)} \lambda_{\max}\left(B(z(\lambda), \lambda)\right)$$

subject to

$$B(z(\lambda), \lambda) < 0, \qquad \|z(\lambda)\| \le R_2,$$

where

$$B(z(\lambda), \gamma) = H(x(\lambda), \lambda) + \sum_i \frac{\partial H}{\partial x_i}\Big|_{x=x(\lambda)} z(\lambda)\Delta\lambda + \frac{\partial H}{\partial \lambda}\Delta\lambda$$

is an approximation of $H(x(\lambda) + z(\lambda)\Delta\lambda, \lambda + \Delta\lambda)$ and $\Delta\lambda$ is the (fixed) increment in λ.

5. Predict $x(\lambda + \Delta\lambda)$ using

$$x^{(0)} \leftarrow x(\lambda) + z(\lambda)\Delta\lambda.$$

6. Set $\gamma \leftarrow \gamma(\lambda + \Delta\lambda)$, $\lambda \leftarrow \lambda + \Delta\lambda$ and go to Step (2).

Observe that Step 2 is always possible because $H(x^{(0)}, \lambda) < 0$, and that Step 4 always has a solution for $\Delta\lambda$ sufficiently small. If Step 4 has no solution, decrease $\Delta\lambda$ and/or increase R_2. Both, steps (2) and (4), are LMIs which can be solved efficiently using standard techniques (Nemirovskii and Gahinet 1994, Boyd and El Ghaoui 1993) and tools for the solution of LMIs are readily available (e.g. Gahinet *et al.* 1995). Constraining the magnitudes of 'dx' and 'z' (via R_1 and R_2) in steps (2) and (4), to be small, allows the LMIs at each iteration, to be a good approximation to their corresponding NMI's. In step (2) instead of 'letting k approach infinity', in practice, it is found to be sufficient to stop the iteration when the maximum eigenvalue of the corresponding NMI is negative.
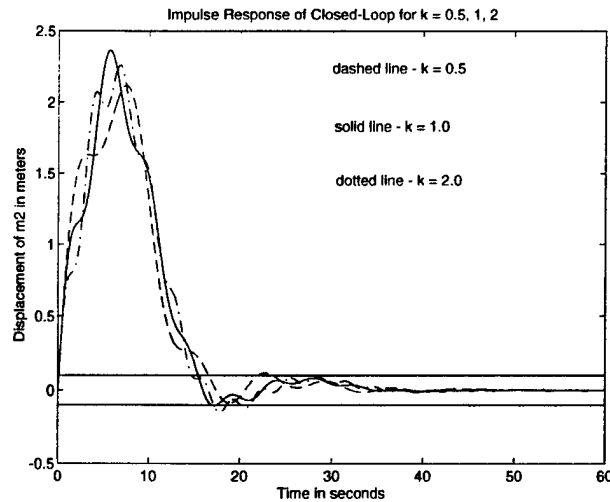
Figure 5.2: Impulse Response

5.3.2 Robust Control Synthesis Using the Popov Multiplier for a Benchmark Problem

To demonstrate the use of the continuation algorithm discussed in the previous section, we consider the two-mass/spring benchmark system shown in Fig. 3.2 with uncertain stiffness k. A control force acts on the body 1 and the position of body 2 is measured, resulting in a noncollocated control problem. This benchmark problem is discussed in detail in Wei and Bernstein (1992).

We desire to design a constant gain linear feedback fixed-structure compensator $K(s)$, such that the closed-loop system is stable for $0.5 < k < 2.0$ and for a unit impulse disturbance at $t = 0$, the performance variable z has a settling time of about 15 s for the nominal system (with $k = k_{nom} = 1$).

Observations

The controller is guaranteed by the theory to be robust for the range $0.5 < k < 2.0$ and this was also verified by a direct search. The settling time for the system was chosen to be the time required for the displacement of mass 2 to reach and stay within the interval [-0.1m, 0.1m]. The controller is seen to satisfy the settling time objectives when connected to the nominal model corresponding to $k = 1$ N/m, as can be seen from the impulse response of the closed-loop system in Fig. 5.2. It

can also be seen that the settling time objective is satisfied for the entire family of plants $(0.5 < k < 2.0)$, which, though not a design requirement, is a very desirable characteristic of the controller.

5.4   Conclusions

The LMI approach to MSSV analysis has been used to develop an approach to robust controller synthesis that is based on stable factors of the multipliers and does not require the multipliers to be restricted to a stable basis. This approach, in general leads to nonlinear matrix inequalities (NMI's), and in special cases to bilinear matrix inequalities (BMIs). An effective continuation algorithm has been developed to solve NMI's (and hence BMIs). The primary computational burden of this algorithm is the solution of a series of linear matrix inequalities (LMIs). The use of this algorithm has been demonstrated by designing a robust controller for a benchmark problem.

CHAPTER 6

FIXED-STRUCTURE NONLINEAR OPTIMAL OUTPUT FEEDBACK STABILIZATION FOR NONLINEAR SYSTEMS

6.1 Introduction

Although the theory for designing linear output feedback controllers is quite mature, nonlinear output feedback controller synthesis remains relatively undeveloped. In numerous real world applications system nonlinearities such as saturation, relay, deadzone, quantization, geometric, and material nonlinearities require nonlinear output feedback controllers. Furthermore, for linear plants with parametric uncertainty and nonquadratic performance criteria, nonlinear controllers exist that generate superior performance over the best linear controller (Haddad *et al.* 1998). In this chapter we develop a fixed-structure controller synthesis framework for nonlinear control. The motivation for fixed-structure nonlinear control theory is to address controller synthesis within a class of candidate nonlinear feedback controller structures. Specifically, control Lyapunov functions are used to provide a controller synthesis framework by assuring global or regional asymptotic stability for an *a priori* fixed class of nonlinear feedback controllers. A specific controller within this class can now be chosen to optimize a given performance functional. Thus, this provides a constructive framework where Lyapunov theory is used to guarantee global or regional asymptotic stability over a class of nonlinear feedback controllers while optimization is performed over the free controller gains so as to minimize a specific performance functional.

It is important to note that the proposed nonlinear controller synthesis framework is quite different from the classical optimal nonlinear control approach predicated on the Maximum Principle. Specifically, the Maximum Principle does not guarantee

asymptotic stability via Lyapunov functions and does not necessarily yield feedback controllers. In contrast, the proposed approach provides a constructive design control Lyapunov function framework for full-state and output feedback control of nonlinear systems using a two-stage optimization framework that guarantees closed-loop stability and optimality with respect to a designer specified performance criterion.

The first stage of this approach is concerned with the synthesis of fixed-structure, state feedback and output feedback, nonlinear control laws and corresponding fixed-structure control Lyapunov functions that increase the domain of attraction of a given nonlinear system about an equilibrium point of the system. The reason for explicitly considering increasing the domain of attraction, as opposed to ensuring global asymptotic stability, is that in practice it may be sufficient to have a controller with an adequately large domain of attraction. Furthermore, for some nonlinear systems global asymptotic stability may not be achievable via nonlinear output feedback (or even state feedback). However, it should be recognized, as will be subsequently demonstrated, that this approach does have the ability to synthesize globally stabilizing control laws.

The second stage of this approach is concerned with finding a fixed-structure nonlinear control law that optimizes an *a priori* chosen performance functional. It is assumed that the first stage, described above, results in a set of (regionally) stabilizing control laws. This second stage then finds a member of this set which optimizes a particular cost function. It is important to note that our nonlinear controllers are *not* predicated on an inverse optimal control problem (Moylan and Anderson 1973, Freeman and Kokotovic 1996, Sepulchre *et al.* 1997, Haddad *et al.* 1998) wherein, in order to avoid the complexity in solving the Hamilton-Jacobi-Bellman equation, a *derived* cost functional as opposed to a *given* cost functional is minimized. Even though inverse optimal controllers may possess indirect robustness guarantees to multiplicative input uncertainty, the performance of the resulting controllers can be arbitrarily poor when compared to the optimal performance as measured by a designer specified cost functional. Furthermore, since such controllers are predicated on Hamilton-Jacobi-Bellman theory they are limited to full-state feedback control.

6.2 Fixed-Structure Controller Design for Nonlinear Systems

In this section we develop a two-step design approach for synthesizing fixed-structure, state and output feedback nonlinear optimal control laws for nonlinear affine systems. First, we focus on the development of stabilizing control laws and then extend our framework to determine stabilizing controllers that optimize a *given* performance functional.

6.2.1 Design of Stabilizing Nonlinear Control Laws

In this section we consider nonlinear affine systems of the form

$$\dot{x}(t) = f(x(t)) + g(x(t))u(t), \quad x(0) = x_0, \quad t \geq 0, \tag{6.1}$$
$$y(t) = h(x(t)), \tag{6.2}$$

where $x \in \mathcal{R}^n$, $u \in \mathcal{R}^m$, $y \in \mathcal{R}^l$, $f : \mathcal{R}^n \to \mathcal{R}$, $g : \mathcal{R}^n \to \mathcal{R}^{n \times m}$, and $h : \mathcal{R}^n \to \mathcal{R}^l$. We assume that $f(\cdot), g(\cdot)$, and $h(\cdot)$ are smooth (C^1 mappings) and $f(\cdot)$ has at least one equilibrium so that $f(0) = 0$ and $h(0) = 0$. In this chapter we focus our attention on systems (6.1) where $f(\cdot)$, $g(\cdot)$ and $h(\cdot)$ are polynomial functions. If $f(\cdot)$, $g(\cdot)$ and $h(\cdot)$ are nonpolynomial, then it follows from the Stone-Weierstrass theorem that they may be approximated by polynomial functions using a Taylor series approximation. In this case, increasing the domain of attraction of the approximated system would increase the domain of attraction of the original system. Hence, this represents an important and fairly general class of nonlinear systems.

To develop a nonlinear controller synthesis framework, we fix the structure of the control Lyapunov function candidate $V(x)$ and the control input $u(t) = \phi(x(t))$ to have the general polynomial form given by

$$V(x) = x^T P_1 x + \sum_{i=2}^{q} (x^T P_i x)^i, \tag{6.3}$$

where $P_1 \in \mathcal{P}^{n \times n}, P_i \in \mathcal{N}^{n \times n}$ for all $i \in \{2, \ldots, q\}$, and

$$u = \phi(y) = \sum_{i_i, i_2, \ldots, i_l} k_{i_1 i_2 \cdots i_l} y_1^{i_1} y_2^{i_2} \cdots y_l^{i_l}, \tag{6.4}$$

where $k_{i_1 i_2 \cdots i_l} \in \mathcal{R}$ for all $i_1 i_2 \cdots i_l$. Now, let θ represent the vector of free parameters corresponding to the sign definite matrices P_i and the controller parameters $k_{i_1, i_2, \cdot, i_l}$,

that is,

$$\theta \triangleq [\text{vec}(P_i),\ k_{i_1 i_2 \cdots i_l}], \tag{6.5}$$

where 'vec' denotes the column stacking operator. In this case, the total derivative of the control Lyapunov function along the trajectories of the closed-loop system is given by

$$\dot{V}(x) = -N(\theta, x) = -x^T B(\theta) x \ - \sum_{i_i, i_2, \dots, i_n} C(\theta)_{i_1 i_2 \cdots i_n}\, x_1^{i_1} x_2^{i_2} \cdots x_n^{i_n}, \quad i_i + i_2 + \cdots + i_n \neq 2, \tag{6.6}$$

where $B(\theta) \in \mathcal{P}^{n \times n}$ and $C(\theta)_{i_1 i_2 \cdots i_n} \in \mathcal{R}$ for all $i_1 i_2 \cdots i_n$.

Next, we use the elements of θ as degrees of freedom to find the controller and control Lyapunov function which maximizes the domain of attraction for the closed-loop system.

Let $\mathcal{I}_e$ denote the set of even nonnegative integers including zero and define the sets ϵ_+^n and δ_+^n as

$$\epsilon_+^n \triangleq \{(i_1, i_2, \cdots, i_n) : i_j \in \mathcal{I}_e,\ j \in \{1, \dots, n\},\ \sum_{j=1}^n i_j \neq 2\}, \tag{6.7}$$

$$\delta_+^n \triangleq \{(i_1, i_2, \cdots, i_n) : j \in \{1, 2, \dots, n\},\ i_j \notin \mathcal{I}_e,\ \sum_{j=1}^n i_j \neq 2\}. \tag{6.8}$$

. Define

$$E(\theta) \triangleq \sum_{i_1 i_2 \cdots i_n} (C(\theta)_{i_1 i_2 \cdots i_n})^2, \quad (i_1, i_2, \cdots, i_n) \in \delta_+^n, \tag{6.9}$$

and denote P_i as $P_i(\theta)$.

Numerical Algorithm for Design of Stabilizing Controller.

For $\epsilon = \epsilon_0$ to 0 in steps of $\Delta \epsilon$

    $\min_\theta \tilde{J}_s(\theta) \triangleq E(\theta) - r_0 \log(\det(P_i(\theta))) - r_0 \log(\det(B(\theta) + \epsilon I))$

    subject to

    $C(\theta)_{i_1 i_2 \cdots i_n} > 0, \quad (i_1, i_2, \cdots, i_n) \in \epsilon_+^n$

end.

For $r = r_0$ to r_f, where r_f is a small constant close to zero, in steps of Δr

$$\min_\theta \tilde{J}_s(\theta) \triangleq E(\theta) - r \log(\det(P_i)) - r \log(\det(B))$$

subject to

$$C(\theta)_{i_1 i_2 \cdots i_n} > 0, \quad (i_1, i_2, \cdots, i_n) \in \epsilon_+^n$$

end.

The constant ϵ_0 is chosen large such that the initial feasible set is large and an initial feasible point θ is easy to find. As $r \to 0$, the solution θ approaches the solution of the actual Optimization Problem.

6.2.2 Design of Optimal Nonlinear Control Laws

Using the approach outlined in the previous subsection, we can design a stabilizing nonlinear controller for a given nonlinear system. This framework can be extended to an optimization problem in which it is desired to minimize a given performance criterion of the form

$$J_p(u, x_0) = \int_0^\infty L(x(t), u(t)) dt, \tag{6.10}$$

where $L : \mathcal{R}^n \times \mathcal{R}^m \to \mathcal{R}$ is not necessarily quadratic. Specifically, given an initial condition x_0 and a stabilizing controller, the closed-loop system may be numerically integrated to obtain the trajectories of $x(t)$ and $u(t)$, $t \geq 0$, and hence $J_p(u, x_0)$ may be evaluated. For asymptotically stable systems the contribution to the integral (6.10) becomes negligible after some finite time. Hence, in practice, it is sufficient to integrate over a finite time interval.

To eliminate the dependence of the synthesized controller on the initial condition x_0, the controller is synthesized to optimize the average of several cost functions corresponding to a number of evenly distributed initial conditions in a specified region of state space. The performance functional (6.10) is therefore replaced by

$$J_{\text{avg}}(u) = \sum_i \frac{1}{\rho(x_{0,i})} J_p(u, x_{0,i}), \tag{6.11}$$

where $\rho : \mathcal{R}^n \to \mathcal{R}_+$ is some function of the distance of $x_{0,i}$ from the origin. This yields a stabilizing controller which is optimal, for the chosen structure of the control law, over a specified region of state space in an average sense.

It is important to restrict the space over which the optimization is carried out to the set of stabilizing controllers since the cost function is defined only in this set.

This results in the following optimization problem:

$$\min_{\theta} \tilde{J}(u) \triangleq w_p J_{\text{avg}}(u) + w_s J_s(u), \tag{6.12}$$

subject to the constraints

$$C(\theta)_{i_1 i_2 \cdots i_n} > 0, \ (i_i, i_2, \ldots, i_n) \in \epsilon_+^n, \tag{6.13}$$

where $J_s(\theta)$ is the cost for the stability part discussed in the previous subsection, and w_p and w_u are constants for the performance and stability part of the cost function, respectively. Since $J_s(\theta)$ effectively acts as a barrier function, w_s should be chosen to be much smaller than w_p.

6.3 Illustrative Numerical Examples

In this section we present several numerical examples to demonstrate the efficacy of the proposed approach. The first example illustrates the application of the design approach for the synthesis of full-state feedback stabilizing nonlinear controllers. The next example discusses the synthesis of output feedback stabilizing nonlinear controllers. The last example presents the design of optimal, output feedback nonlinear control laws. An optimization problem must be solved in the performance phase of the design, and at each step of the continuation algorithm, in the stability phase of the design. All optimization problems are solved using a constrained BFGS algorithm (Fletcher 1987) in the Matlab Optimization Toolbox (Grace 1992).

Example 1 (Full-State Feedback). Consider the open-loop unstable nonlinear system

$$\dot{x}_1 = -x_1 + x_2 x_1^2, \tag{6.14}$$

$$\dot{x}_2 = x_2 + u. \tag{6.15}$$

Note that for this system the domain of attraction of the open-loop system consists of a single point, namely, the origin. We assume that all states are measurable and choose a quadratic control Lyapunov function $V(x) = x^T P x$, and a controller of the form

$$u = k_{10} x_1 + k_{01} x_2 + k_{20} x_1^2 + k_{11} x_1 x_2 + k_{02} x_2^2 + k_{30} x_1^3 + k_{21} x_1^2 x_2 + k_{12} x_1 x_2^2 + k_{03} x_2^3. \tag{6.16}$$

Using a variety of initial conditions and constraining all the optimization variables between -10 and 10, we arrive at the following solution:

$$V(x) = 10x_1^2 + 10x_2^2,$$

$$u = -10x_2 - x_1^3 - 2x_1^2 x_2 - 1.63x_2^3.$$

It can be easily verified that all constraints are satisfied. In addition, for this particular example, $E = 0$, so that E achieves its minimum value. Hence, in this case the domain of attraction is $\mathcal{R}^n$, i.e., the controller obtained is globally asymptotically stable. This shows that if appropriate forms are chosen for the control Lyapunov function and the controller, global asymptotic stability may be obtained.

Example 2 (Static Output Feedback). It is now assumed that in Example 1 only the state x_2 is measurable. A quadratic control Lyapunov function $V(x) = x^T P x$ is chosen as in Example 1, and the structure

$$u = k_{01}x_2 + k_{02}x_2^2 + k_{03}x_2^3, \tag{6.17}$$

is chosen for the controller.

Using a variety of initial conditions and constraining the optimization variables between -5 and 5, we arrive at the following solution:

$$V(x) = 0.353x_1^2 + 5x_2^2,$$

$$u = -5x_2 - 0.09x_2^3.$$

The minimum value of E achieved is 0.5 and k_{01} achieves its lower bound, i.e., $k_{01} = -5$. The fact that the minimum value of E is not close to zero indicates that global stability has not been achieved in this case. However, simulations indicate that the domain of attraction has been increased, as shown in Figure 6.1. Furthermore, the fact that the variable k_{01} attains its lower bound, strongly suggests that the domain of attraction may be further increased by decreasing this lower bound. Constraining all the variables between -10 and 10 we get the following solution:

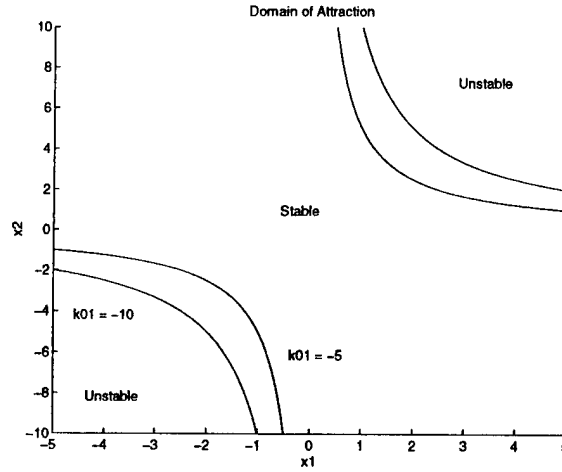$$V(x) = 0.005x_1^2 - 0.00016x_1x_2 + 10x_2^2,$$

Figure 6.1: Domain of Attraction

$$u = -10x_2 - 0.0048x_2^3.$$

The minimum value of E achieved is now 10^{-4} and once again we see that the variable k_{01} attains its lower bound. The domain of attraction of the system has been further increased as can be seen in Figure 6.1. Hence, output feedback (i.e., partial state feedback) is unable (for the assumed controller structure) to provide global asymptotic stability. However, the domain of attraction of this system may be increased at the expense of increased controller authority.

Example 3 (Performance via Output Feedback). It is desired to find an optimal output feedback controller for the nonlinear system given in Example 1, that minimizes the performance functional

$$J_p(u, x_0) = \int_0^\infty (x_1^2 + x_2^2 + u^2)dt, \tag{6.18}$$

where x_0 denotes the initial condition of system states. and the output is the state x_2. A controller structure of the form

$$u = k_{01}x_2 + k_{03}x_2^3, \tag{6.19}$$

is chosen based on the results of Example 2. The solution

$$u = -10x_2 - 0.0048x_2^3, \tag{6.20}$$

of Example 2 is chosen as the starting point for the optimization procedure. To eliminate the dependence of the synthesized controller on the initial condition x_0 the performance functional is modified as shown earlier. The initial conditions x_0 are chosen as the four corners of several symmetric squares about the origin, viz., (z, z), $(z, -z)$, $(-z, z)$ and $(-z, -z)$, for $z = 0.5, 1.5$ and 3. That is, a total of 12 initial conditions are chosen from the domain of attraction shown in Figure 6.1. The closed-loop system is integrated to a time $t = 20$ seconds, since it is observed that the contribution to the performance functional becomes negligible after this time. The optimal controller (for the chosen controller structure) is given by

$$u = -10x_2 - 0.7x_2^3. \tag{6.21}$$

In this case it is found that the optimal controllers produced for each of the initial conditions chosen are quite different. Specifically, for points (z, z) and $(-z, -z)$ the optimal controller gains are high especially for points near the boundary of the domain of attraction (see Figure 6.1), while for points $(-z, z)$ and $(z, -z)$ the optimal controller gains are low. Hence, in this case, while the controller is optimal in an average sense for the chosen controller structure, the optimality is far from global. This is inevitable and is the price that must be paid for the unavailability of measurements of the state x_1.

6.4 Conclusion

A constructive procedure for the synthesis of fixed-structure, optimal, state and output feedback nonlinear controllers has been developed. The first step of this approach synthesizes stabilizing controllers which increase the domain of attraction of the closed-loop nonlinear system. The second step determines the member of a set of stabilizing controllers that optimizes a designer specified performance functional. Several examples have been used to demonstrate the efficacy of the proposed design procedure. The examples suggest, that if an appropriate form is chosen for the control Lyapunov function and the nonlinear controller, both global stability and global optimality may be obtained. This design method is applicable to general nonlinear affine systems, since they can always be approximated by polynomial systems. Furthermore, it may also be be applied directly (i.e., without polynomial approximation)

to certain classes of general nonlinear affine systems, for which the structure of control Lyapunov functions and controllers are known *a priori*, or at least an educated guess can be made based on prior experience.

CHAPTER 7

AN OBJECT-ORIENTED APPROACH TO SEMIDEFINITE PROGRAMMING

7.1 Introduction

Object-oriented design and programming has been a major theme in software engineering in recent years. Traditional design, the main software design paradigm until about the mid 1980s, concentrates on the actions that a system has to take and decomposes the system into separate units or modules according to their functionalities. In object-oriented design a system to be modeled is viewed as a collection of objects, each of which has its own attributes and the operations performed on an object or functions acting on an object are also defined in one syntactic unit. Objects communicate by passing messages or by calling functions from other objects which provide services. Object-oriented design is developing an object-oriented model of a system and can be realized (implemented) by object-oriented programming using languages such as C++, FORTRAN 90, or Smalltalk.

The advantages of object-oriented design and programming have been described widely elsewhere (Booch, 1994). A short summary will be provided here. First, an object is an independent entity that is encapsulated in one syntactic unit. The definition of an object consists of the definition of the attributes of the object along with operations that can be performed on the object and the services or function calls provided by the object. Encapsulation hides the implementation details of an object and makes the program easier to read and modify. Any subsequent change to the program can be localized, making the resulting program more easily maintained.

The second advantage is information hiding. Definitions of an object which need not be known to other objects are inaccessible to other objects, preventing them

from being changed accidentally. In other words, information hiding makes implementation details of an object inaccessible to other objects. However, the designer has the freedom to decide what to hide and what not to hide.

The third advantage is code reuse. Inheritance enables the definition of a new object, which can be viewed as a subclass of an existing object, without having to repeat some of the details. The new object can inherit attributes or operations from its ancestor. Inheritance is one way to support reuse of existing objects. There are different kinds of reuse in object-oriented programming; inheritance is only one of them.

The object-oriented technology has been accepted in the software engineering community as a better approach to develop maintainable software for large and complex systems. The same should apply to large scale numerical computation applications. Even though efficiency has been the major concern for most numerical computation applications, for large scale multidisciplinary computations, maintainability and reusability of software components may become the more important consideration.

One of the most popular object-oriented programming languages is C++ (Stroustrup 1991), which is used to implement the algorithm of this paper. Some of the reasons why C++ is so widely used are upward compatibility with C, design emphasis on efficiency and performance, and the availability of many useful libraries and tools. For instance, the Gnu C++ compiler and other tools are available on a wide range of platforms and provide good performance, programming environments, and reasonable compliance with ANSI standards.

There are many available libraries such as IML++ (Dongarra *et al.* 1994b), SparseLib++ (Dongarra *et al.* 1994a, Pozo *et al.* 1994), STL (Stepanov and Lee 1993, Musser and Saini 1996), and others which emphasize numerical computation. One notable package is LAPACK++, developed by Dongarra *et al.* (1994a), which is a C++ interface to LAPACK and BLAS. Dongarra *et al.* (1994a) has shown that performance of programs using the package is comparable to calling LAPACK and BLAS directly, and can at the same time reap the benefits of object-oriented programming.

This chapter contains the result of object-oriented design and implementation of an algorithm for semidefinite programming. Semidefinite programming refers to minimizing a linear function subject to a linear matrix inequality (Vandenberghe and S. Boyd 1996a). That is,

$$\min_{x \in \mathbf{R}^m} c^T x \qquad (7.1)$$

$$\text{subject to} F(x) \geq 0,$$

where

$$F(x) \equiv F_0 + \sum_{i=1}^{m} x_i F_i,$$

$c \in \mathbf{R}^m$, and $F_0, \ldots, F_m \in \mathbf{R}^{n \times n}$ are symmetric matrices. $F(x) \geq 0$ means that $F(x)$ is positive semidefinite.

Many problems in controls engineering can be cast in terms of a semidefinite programming problem (Vandenberghe and Boyd 1996a). Since a semidefinite programming problem is a convex optimization problem, which can be solved by interior point methods (Nesterov and A. Nemirovsky 1994), it has attracted the attention of many researchers in interior point methods. There is a C implementation of a primal-dual algorithm for solving the semidefinite programming problem (Vandenberghe and Boyd 1996b). A C++ implementation of that primal-dual algorithm for the semidefinite programming problem is developed here to explore the possible benefits of object-oriented design. Because of the similarity of the primal-dual algorithm with other interior point algorithms for solving the semidefinite programming problem, the design and implementation methodology developed here can be easily modified and applied to other interior point algorithms.

The performance of a C++ implementation of the primal-dual algorithm for semidefinite programming is compared with the existing C implementation of the same algorithm from Vandenberghe and Boyd (1996b). While the CPU times of the two implementations are comparable to each other, the C++ version offers the advantages mentioned earlier in this section.

7.2   The C++ implementation

The primal-dual algorithm is used in this chapter for solving the semidefinite programming problem. This algorithm is described in detail in Vandenberghe and

S. Boyd (1996a) and will not be repeated here. The program is built upon the LAPACK++ v1.0 package, especially the LaVectorDouble, LaGenMat, LaSymm-Mat classes, and BLAS++ is used extensively. Several special purpose routines are added to the LAPACK++ package to accommodate the primal-dual algorithm for semidefinite programming. The program also uses the iterator object in STL (Standard Template Library) (Stepanov and Lee 1993, Musser and Saini 1996) to traverse arrays of objects.

The first major difference between the C and C++ implementations is the way the initial data is read in. Unlike C/Fortran style subroutines, in which one can pass a pointer/address for a piece of storage and let the subroutine split the storage into pieces to get the data, C++ objects' constructors have no such scheme. Initialization is done by reading a data file.

The second major difference is that because C++'s objects are higher level abstractions, the implementation in C++ is less dependent upon pointer arithmetic for doing the same computation. There is overhead associated with this higher level of abstraction, but we will show that the effect on performance is negligible.

7.3 Comparison and discussion of results

Two sets of data are obtained by randomly generating all the matrices F_0, F_1, $\cdots$, F_m, and the vector c. Strictly feasible initial points x_0 and Z_0 are also generated. The timing results are shown in Table 1. All the timings are done on a HP 712/60 workstation. Both the C implementation from Vandenberghe and S. Boyd (1996b) and the C++ implementation are compiled using the Gnu C/C++ compiler version 2.7.2 with the same compiler options. It is clear from Table 7.1 that the performance penalty for using C++ is only a few percent and decreases as the problem size increases.

Table 7.1. Comparison of implementations.

	Example 1, $n = 40$	
	C++ implementation	C implementation
m	time (sec)	time (sec)
20	4.7	4.4
30	6.9	6.5
	Example 2, $n = 100$	
50	229	222
75	390	381

7.4 Conclusions

We have shown an objected-oriented design and implementation of a semidefinite programming algorithm. Even though object-oriented technology is being used more and more widely in industry now, there are not many realistic applications to numerical computation. The programming environments and tools seem to be mature enough to apply this new methodology, and the performance seems to be comparable to a non-object-oriented implementation.

However, there are other considerations that have to be taken into account when applying object-oriented technology. First, it takes time and effort to learn the new methodology. Second, it is not a trivial task to set up the environment: compiling all the C++ packages, and verifying that they work correctly, especially when most of the C++ packages for numerical computation are still in the testing stage. Third, the resulting code size, i.e., the size of the C++ executable, is about 2.5 times that of the C executable. With continuing development of object-oriented technology and of compilers for object-oriented languages, the second problem, will likely be alleviated. The hardware considerations of the third problem are becoming less of a hindrance with the advances in the computer industry. We do believe that the benefits of using object-oriented methodology outweight the currently existing disadvantages.

The design and analysis in this research can be generalized to apply to object-oriented design and implementation of other interior point algorithms which use potential reduction to find the optimum.

CHAPTER 8

COST-EFFECTIVE PARALLEL PROCESSING FOR H_2/H_∞ CONTROLLER SYNTHESIS

8.1 Introduction

H^2/H^∞ mixed-norm controller synthesis is an important and interesting technique in modern control design which provides the means for simultaneously addressing H^2 and H^∞ performance objectives. In practice such controllers provide both nominal performance (via suboptimal H^2) and robust stability (via H^∞). Hence mixed-norm synthesis provides a technique for trading off performance and robustness, a fundamental objective in control design.

The H^2/H^∞ mixed-norm problem has been addressed in a variety of settings. One treatment utilized an H^2 cost bound as the basis for an auxiliary nonconvex constrained minimization problem, which is very difficult without the global convergence of homotopy methods. A successful homotopy algorithm based on the Ly form parametrization has been developed (Ge *et al.* 1994).

The H^2/H^∞ control design algorithms will be used for controller design of systems such as the four disk system of Cannon and Rosenthal (1984). This system is especially representative of the type of vibration control problems that arise in industrial problems involving rotating turbomachinery. H^2/H^∞ design will be used to develop controllers that are robust with respect to the unmodeled dynamics and also guarantee a certain measure of nominal performance.

It should be mentioned that H^2/H^∞ theory has been used in Haddad (1993g,1994a,1994b) to develop complex structured singular value synthesis (CSSV) formalisms that *a priori* fix the structure of both the D-scales and the controller. Hence, an extension of the algorithms here will enable fixed-structure CSSV controller synthesis that blends H^2 and H^∞ performance objectives.

Practical applications often lead to large dense systems of nonlinear equations which are time-consuming to solve on a serial computer. For these systems, parallel processing may be the only feasible answer to these problems. One economical way of achieving parallelism is to utilize the aggregate power of a network of heterogeneous serial computers. In industrial environments where interactive design is often the practice, the parallel code can be easily incorporated into interactive software such as MATLAB or Mathematica with proper setup of the network computers. To the engineering users the design environment is the same except that the computation is faster.

The most expensive part of the homotopy algorithm is the computation of the Jacobian matrix, which can be parallelized easily to run across an Ethernet network with little modification of the original sequential code, and which has relatively large task granularity. There is a trade-off between the programming effort and the speedup of the parallel program. To obtain a better speedup, other parts of the homotopy algorithm, such as finding the solution to the Riccati equations and the QR factorization to compute the kernel of the Jacobian matrix, need to be parallelized as well.

In this study a homotopy algorithm for the H^2/H^∞ controller synthesis problem is parallelized to run on a network of workstations using PVM (Parallel Virtual Machine) and an Intel Paragon parallel computer, under the philosophy that as few changes as possible are to be made to the sequential code while achieving an acceptable speedup. The parallelized computation is that of the Jacobian matrix, which is carried out in the master-slave paradigm by functional parallelism, that is, each machine computes a different column of the Jacobian matrix with its own data. Unless the Riccati equation solver is parallelized, there is a large amount of data needed for each slave process at each step of the homotopy algorithm. To avoid sending too many large messages across the network or among different nodes on the Intel Paragon, all slave processes repeat part of the computation done by the master process, which therefore decreases the speedup of the parallel computation.

The speedups of the parallel code are compared as the number of workstations increases or the number of nodes increases on a Intel Paragon and as the size of the problem varies. A reasonable speedup can be achieved using an existing network

of workstation compared to that of using an expensive parallel machine, the Intel Paragon. It is demonstrated that for a large problem, the approach of using a network of workstations to parallelism is feasible and practical, and provides an efficient and economical computational method to parallelize a homotopy based algorithm for H^2/H^∞ controller synthesis in a workstation-based interactive design environment.

8.2   The homotopy algorithm

The H_2/H_∞ controller synthesis problem is described in detail in Haddad and Bernstein (1990), and is not repeated here. It suffices to say that the end problem is a minimization problem subject to a Riccati equation constraint. The necessary conditions for an extremum result in a root finding problem which is solved using a homotopy algorithm.

The homotopy zero curve tracking algorithm (which is a standard globally convergent probability-one homotopy algorithm (Watson et al. 1987b) is

1. Set $\lambda := 0$, $\theta := \theta_0$.

2. Evaluate the homotopy map $\hat{\rho}$ and the Jacobian of the homotopy map $D\hat{\rho}$.

3. Predict the next point $Z^{(0)} = (\theta^{(0)}, \lambda^{(0)})$ on the homotopy zero curve using, e.g., a Hermite cubic interpolant.

4. For $k := 0, 1, 2, \cdots$ until convergence do $Z^{(k+1)} = Z^{(k)} - [D\hat{\rho}(Z^{(k)})]^\dagger \hat{\rho}(Z^{(k)})$, where $[D\hat{\rho}(Z)]^\dagger$ is the Moore-Penrose inverse of $D\hat{\rho}(Z)$. Let $(\theta_1, \lambda_1) = \lim_{k \to \infty} Z^{(k)}$.

5. If $\lambda_1 < 1$, then set $\theta := \theta_1$, $\lambda := \lambda_1$, and go to step 2).

6. If $\lambda_1 \geq 1$, compute the solution $\bar{\theta}$ at $\lambda = 1$.

8.3   The parallel algorithm

The major part of the computation in Step 2) is that of the Jacobian matrix. The number of variables in this formulation is $n_c(m + l) + 1$ including λ. Each column of the Jacobian matrix corresponds to the derivative of the homotopy map

with respect to one variable and requires the solution of two Lyapunov equations (Ge *et al.* 1994). Therefore the time complexity of the Jacobian matrix computation is $O(n_c(m+l)(n+n_c)^3)$. The Bartels and Stewart algorithm finds the real Schur form of $\tilde{A}$ or $\tilde{A}^T$ depending on the Lyapunov equation. Unnecessary factorization can be avoided if the previous factorization results from the computation of $\hat{\rho}$ and $D_\lambda \hat{\rho}$ are used.

The primary goal of this study is to make use of the existing code and to achieve reasonable parallel efficiency economically. The only part of the algorithm that is parallelized is the Jacobian matrix computation in Step 2). To utilize existing computer resources such as a network of workstations, the software package PVM (Parallel Virtual Machine) is used to provide the distributed computing capabilities.

The parallel algorithm follows the master-slave paradigm. The master sends the index of the column of the Jacobian matrix to be computed to a slave. The slave computes the corresponding column of the Jacobian, sends the column back to the master, and waits for the next index from the master to arrive. After receiving a column of the Jacobian, the master sends another index to the idle slave. In the implementation for the Intel Paragon, asynchronous send is used whenever possible to speed up the communication.
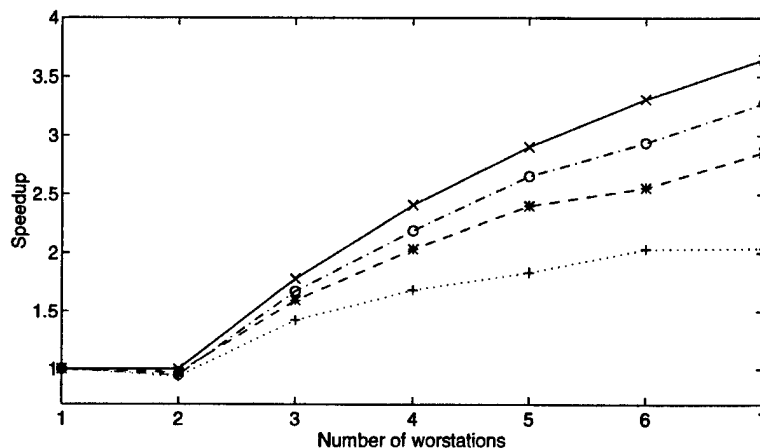

Figure 8.1: Speedup with master and slaves on different machines

When the algorithm is implemented on a network of workstations, the modification to the original sequential source code consists of three parts: the first one is to spawn slave processes and set up the communication links between the master

and the slaves; the second is to extract a slave program from the original code and at the same time simplify the master program; the last is to add a mechanism to guarantee correct communication between master and slaves. The first part consists of standard PVM operations, while the second is more problem oriented. To decrease communication, each slave process repeats part of the computation of $\hat{\rho}$ and $D_\lambda\hat{\rho}$ so as to minimize message passing through the network. There is no loss of efficiency since the master is also computing the same quantities. The slave program consists of mainly the original subroutines with additional code for communication.

For the implementation on the Intel Paragon, the modification of the original code is even simpler. There is no need for a separate slave program if control statements use node identification properly. The parent process run on an Intel Paragon always gets node number 0 while other nodes are numbered bigger than 1 and bigger. The statement *if node_number* == 0 precedes the code that is to be executed by the master and an *else* following the previous statement will precede the code to be executed by the slave. The other modification to the original code is similar to the implementation using PVM. Asynchronous send is used whenever possible. A *wait* is used later when the data is needed, to insure correct communication between the master and the slaves.
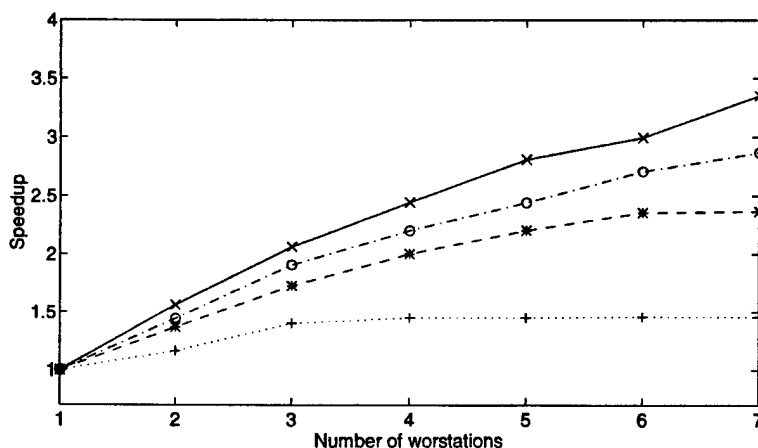
Figure 8.2: Speedup when one slave is on the master machine

8.4 Conclusions

Parallelizing the Jacobian matrix computation in a homotopy algorithm reduces the execution time and is economical, especially for large problems. Acceptable speedups are obtained for a PVM implementation on a network of workstations. The approach can be applied to real industrial design environments to reduce controller design time and effectively utilize existing workstation networks. Compared to using real parallel computers, the approach of utilizing a network of workstations is much more cost-effective.

CHAPTER 9

ROBUSTNESS ANALYSIS IN THE DELTA-DOMAIN USING FIXED-STRUCTURE MULTIPLIERS

The development of mixed (i.e., real and complex) structured singular value analysis and related multiplier-based analysis results (Balakrishnan *et al.* 1994, Fan *et al.* 1991, Haddad *et al.* 1992, Haddad and Bernstein 1995a, 1993, Haddad and Kapila 1995af, How and Hall 1993, Ly *et al.* 1994, Safonov and Chiang 1993, Young 1996) have made a significant impact on the ability of engineers to analyze and design controllers for uncertain systems in the presence of mixed (i.e., real and complex) uncertainty. These results are very powerful and should eventually make a large impact on control engineering practice. However, with the notable exceptions of Haddad and Bernstein (1993), Haddad and Kapila (1995af), Tchernychev and Sideris (1996), to date most of these results have focused on the analysis and synthesis of continuous-time systems. Nevertheless, it is the overwhelming trend to implement controllers digitally.

Interfacing a continuous-time system with a digital computer results in an equivalent discrete-time system. Furthermore, in practice bounds on modeling uncertainty are usually obtained via identification experiments in which the resulting system models are discrete-time. In this case, it is more natural to represent the nominal model of the system and its uncertainty in discrete-time. As is well known, the highest performing discrete-time control laws are developed by directly using a discrete-time representation of the system as the design model.

The intent of the research presented in this chapter is to further advance mixed structured singular value analysis for discrete-time systems. The results are developed using the delta-operator representation of a discrete-time system (Goodwin *et al.* 1992, Middleton and Goodwin 1986, 1990) since it has clear advantages over

the standard forward-shift representation. In particular, the delta-domain represen-tation avoids the numerical ill-conditioning inherent in the use of the forward-shift representation, helps to unify the continuous-time and discrete-time results from a theoretical perspective, can allow the same software algorithms to be used for the analysis and synthesis of control systems for both continuous-time and discrete-time systems, and provides improved performance over standard forward-shift represen-tations when using finite wordlength computation.

This chapter begins by developing frequency-domain robustness analysis tests involving frequency-dependent multipliers. These tests involve strictly generalized positive real and strictly positive real conditions. Hence, in order to develop state space robustness tests, emphasis is placed on the development of state space linear matrix inequality (LMI) and Riccati equation tests for the positive real and gener-alized positive real conditions. By proper construction of the stability multipliers, LMI robustness tests are then developed. It should be noted that these multiplier constructions differ significantly from those given in Tchernychev and Sideris (1996) when both are viewed in either the z-domain or the delta-domain. In fact, when viewed in the delta-domain (as is done here), unlike the multipliers in Tchernychev and Sideris (1996), the multipliers here collapse to the continuous-time multipliers of Collins *et al.* (1997a), Ly *et al.* (1994) as $T \to 0$. The results are then specialized to the case of delta-domain Popov-type multipliers, which because of their simplicity, are particularly useful for robust control law design. Note that in general, state space tests may be used to avoid the frequency-dependent discontinuities that may occur when applying frequency-domain robustness tests (Young 1996). In addition, they can be used as the basis for robust, fixed-architecture control design (Collins *et al.* 1997a, 1997c, Chiang and Safonov 1992, Goh *et al.* 1994a, 1994b, How *et al.* 1994a, 1994b, 1996, Safonov *et al.* 1994).

9.1 Multiplier Methods in the Robustness Analysis of Delta-Domain Systems

In this section we consider the standard uncertainty feedback configuration given in Figure 9.1, where $G(\gamma) \in \mathcal{C}^{m \times m}$ is asymptotically stable and $G(\gamma) \sim \left[\begin{array}{c|c} A & B \\ \hline C & D \end{array} \right]$. It

is assumed that the uncertainty $\Delta \in \mathcal{C}^{m \times m}$ belongs to the block-diagonal uncertainty set

$$\Delta_{\text{bs}} \triangleq \{\Delta = \text{block-diag}(\Delta_1, ..., \Delta_p) : \Delta_i \in \mathcal{I}_i, \ i = 1, ..., p\}, \qquad (9.1)$$

where $\mathcal{I}_i \subseteq \mathcal{C}^{k_i \times k_i}$, $\sum_{i=1}^p k_i = m$, denotes the internal structure of the uncertainty block Δ_i, $i \in [1, \ldots, p]$.
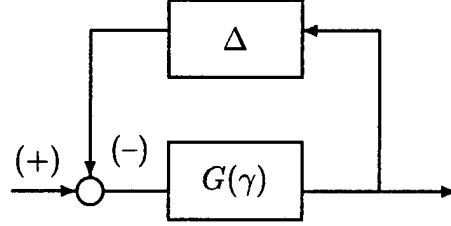

Figure 9.1: Standard Uncertainty Feedback Configuration

Theorem 9.1. Suppose $\hat{G}(\gamma) \triangleq G(\gamma)[I + M_1 G(\gamma)]^{-1}$ is asymptotically stable and $M(\gamma)$ is the multiplier. If there exists $P > 0$ such that

$$\begin{bmatrix} -TA_a^T P A_a - A_a^T P - P A_a & -TA_a^T P B_a - P B_a + C_a^T \\ -TB_a^T P A_a - B_a^T P + C_a & -TB_a^T P B_a + D_a + D_a^T \end{bmatrix} > 0, \qquad (9.2)$$

where (A_a, B_a, C_a, D_a) is the state space realization of $M(\gamma)\hat{G}(\gamma)$, and T is the sampling period, then the negative feedback interconnection of $G(\gamma)$ and Δ is asymptotically stable for all $\Delta \in \Delta$.

Theorem 9.2. Suppose $\hat{G}(\gamma) \triangleq G(\gamma)[I + M_1 G(\gamma)]^{-1}$ is asymptotically stable. If there exists $P > 0$ and $R > 0$ such that

$$D_a + D_a^T - TB_a^T P B_a > 0, \qquad (9.3)$$

$$\begin{aligned} 0 &= TA_a^T P A_a + A_a^T P + P A_a \\ &\quad [B_a^T(TA_a + I) - C_a]^T (D_a + D_a^T - TB_a^T P B_a)^{-1} [B_a^T(TA_a + I) - C_a]^T \quad (9.4) \\ &\quad + R. \qquad (9.5) \end{aligned}$$

where (A_a, B_a, C_a, D_a) is the state space realization of $M(\gamma)\hat{G}(\gamma)$, and T is the sampling period, then the negative feedback interconnection of $G(\gamma)$ and Δ is asymptotically stable for all $\Delta \in \Delta$.

9.2 Illustrative Numerical Example

To illustrate the delta-domain robustness analysis framework presented in the previous sections, we consider the two mass/spring benchmark system with uncertain stiffness k. For a detailed discussion of this benchmark problem, refer to Collins (1997a) and the references therein.

Using Popov analysis (Haddad 1995a) this controller guarantees robust stability for the range $0.334 \leq k \leq 2.166$ which corresponds to the actual stability margin with $\eta = \eta_{\text{achieved}} = 0.916$.

Next we compare the q-domain (i.e., forward shift representation) and δ-domain robust stability tests for the Popov-type multipliers

$$M(\gamma) = H + N\frac{2\gamma}{2 + \gamma T}, \tag{9.6}$$

$$M(\gamma) = H + N\frac{\gamma(2 + \gamma T)}{2(1 + \gamma T)}, \tag{9.7}$$

$$M(\gamma) = H + N\frac{\gamma}{1 + \gamma T}, \tag{9.8}$$

where H and N are real and diagonal and $H > 0$. It can be shown that these are compatible multipliers. These will be referred to as Multipliers 1, 2, and 3, respectively.

The q-domain equivalent for a fixed sampling period T of Multipliers 1, 2, and 3 were obtained by simply replacing $M(\gamma)$ with $M(z)$ where $z = 1 + \gamma T$. In order to simplify the continuous-time to discrete-time conversions and considering the fact that we are interested in small sampling periods, we neglect higher-order terms in the sampling period associated with the uncertainty ΔA_{p}. This yields

$$A_{\text{p}q}(\Delta) = e^{TA_{\text{p}}} + T\Delta A_{\text{p}}, \qquad B_{\text{p}q} = \int_0^T e^{A_{\text{p}}(T-\tau)}B_{\text{p}}\mathrm{d}\tau,$$

and

$$A_{\text{p}\delta}(\Delta) = \frac{1}{T}(e^{TA_{\text{p}}} - I) + \Delta A_{\text{p}} + \frac{T}{2}(A_{\text{p}}\Delta A_{\text{p}} + \Delta A_{\text{p}}A_{\text{p}}), \qquad B_{\text{p}\delta} = \frac{1}{T}B_{\text{p}q}.$$

For the q and δ conversions C_{p} remains unchanged. Similar conversions can be obtained for the controller. Note that the δ-domain representation of the plant collapses to the continuous-time representation as the sampling period $T \to 0$.

The robustness tests for the q-domain and δ-domain are formulated as LMI feasibility problems with varying sampling periods for Multipliers 1, 2, and 3. Figure 9.2 shows the comparison of the allowable uncertainty predictions for the q-domain and δ-domain using Multipliers 1, 2, and 3. It is clear from the figure that as $T \to 0$ the robustness tests for the δ-domain representation continue to track the continuous-time robustness analysis predictions while the q-domain formulation fails to recover the continuous-time robustness boundaries.
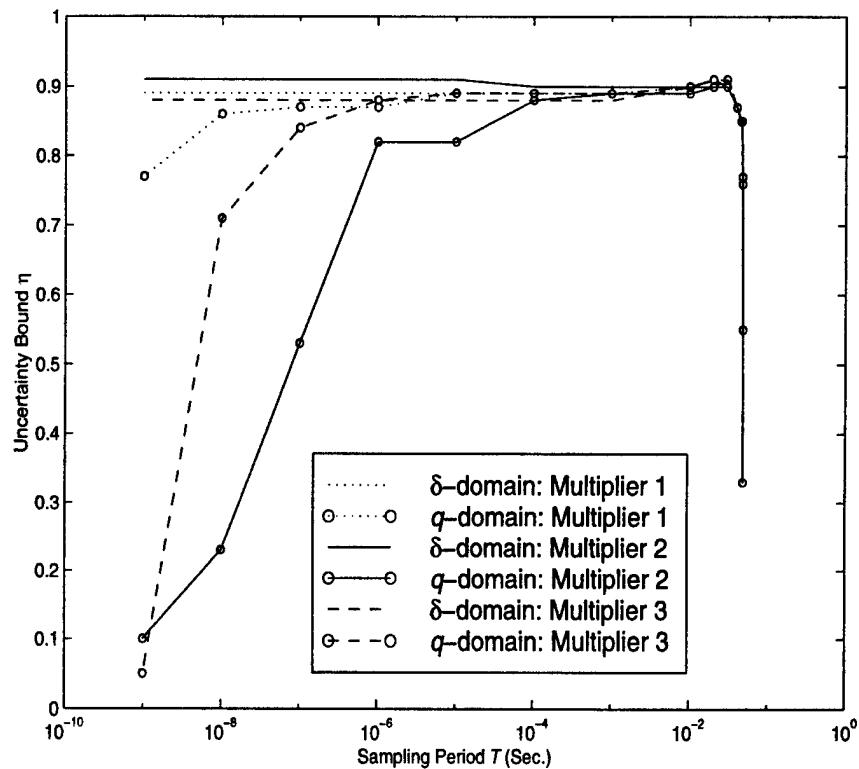

Figure 9.2: Comparison of the q-domain and δ-domain robustness predictions as a function of sampling period T

9.3 Conclusion

A general theory for the robustness analysis of delta-domain systems using multiplier theory was presented. The results are in terms of delta-domain generalized positive real conditions and are analogous to those for continuous-time systems. To

develop state space robustness analysis tests, LMI and Riccati equation characterizations of generalized positive real conditions for delta-domain systems were developed. Subsequently, for the special case of Popov-type multipliers, both LMI and Riccati equation robustness tests were developed. The results were illustrated with a numerical example. A significant contribution of the results was the development of the Popov-type multipliers since the z-domain version of these multipliers (with the exception of the Popov-Tsypkin multiplier) have not appeared in the existing literature.

REFERENCES

[1] ADAMS, R. J. and BANDA, S. S., 1993, Robust flight control design using dynamic inversion and structured singular value synthesis. *IEEE Transactions on Control Systems Technology*, **1**, 80-92.

[2] ALLGOWER, E. L. AND GEORG, K., 1990, *Numerical Continuation Methods* (New York: Springer-Verlag).

[3] ANDERSON, B. D. O. and MOORE, J. B., 1968, Algebraic structure of generalized positive real matrices. *SIAM Journal of Control*, **6**, 615-624.

[4] ANDERSON, B.D.O AND VONGPANITLERD, S., 1973, *Network Analysis and Synthesis: A Modern Systems Theory Approach* (Eaglewood Cliffs, NJ: Prentice Hall).

[5] B. D. O. Anderson and Liu, Y., 1989, Controller reduction: concepts and approaches. *IEEE Transactions on Automatic Control*, **34**, 802-812.

[6] ANDERSON, B. D. O. AND MOORE, J. B., 1991, *Optimal Control, Linear Quadratic Methods* (Prentice-Hall, Englewood Cliffs, NJ).

[7] APKARIAN, P., CHRETIEN, J. P., GAHINET, P., and BIANNIC, J. M., 1993, μ synthesis by $D - K$ iteration with constant scaling. *Proceedings of the American Control Conference*, San Francisco, CA, pp. 3192-3190.

[8] BALAKRISHNAN, V., HUANG, Y., PACKARD, A., and DOYLE, J., 1994, Linear matrix inequalities in analysis with multipliers. *Proceedings of the American Control Conference*, Baltimore, MD, pp. 1228-1232.

[9] BALAS, G. J., CHU, C., and DOYLE, J. C., 1989, Vibration damping and robust control of the JPL/AFAL experiment using μ synthesis. *Proceedings of the Conference on Decision and Control*, Tampa, FL, pp. 2689-2694.

[10] BARTELS, R. H., and STEWART, G. W., 1972, Solution of matrix equation $AX + XB = C$. *Comm. of the ACM*, **15**, 820-826.

[11] BENJAMIN/CUMMINGS, 1994, *Object Oriented Design with Applications, second edition*, Redwood City, CA, 1994.

[12] BERNSTEIN, D. S., and HADDAD, W. M., 1989, LQG control with an H_∞ performance bound: A Riccati equation approach. *IEEE Transactions on Automatic Control*, **AC-34**, 293-305.

[13] BESELER, CHOW AND MINTO, 1992, A feedback descent method for solving constrained LQG control problems. *Proceedings of the American Control Conference*, pp. 1044-1048.

[14] BODE, H. W., 1945, *Network Analysis and Feedback Amplifier Design* (Von Nostrand, Princeton, NJ).

[15] BOYD, S., and EL. GHAOUI, L., 1993, Method of centers for minimizing generalized eigenvalues. *Linear Algebra Appl.,*, special issue on Numerical Linear Algebra Methods in Control, Signals and Systems, **188**, 63-111.

[16] BOYD, S., EL GHAOUI, L., FERON, E., and BALAKRISHNAN, V., 1994, Linear matrix inequalities in systems and control theory. *SIAM*, Philadelphia.

[17] BOYD, S., VANDENBERGHE, L., and GRANT, M., 1994g, Efficient convex optimization for engineering design. *Proceedings of the IFAC Symposium on Robust Control Design*, Rio de Janeiro, Brazil, Sept., 14-23.

[18] BRAATZ, R., AND MORARI, M., 1992, Robust control for a noncollocated spring-mass system. *Journal of Guidance, Control, and Dynamics*, **15**, 1103-1110.

[19] BRYSON A. E., and HO, Y., 1975, *Applied Optimal Control* (Hemisphere Publishing Corporation).

[20] BUSCHEK, H. and CALISE, A., 1997, μ controllers: Mixed and Fixed *Journal of Guidance, Control, and Dynamics*, **20**, 34-41.

[21] CANNON, R. H. and ROSENTHAL, D. E., 1984, Experiments in control of flexible structures with noncolocated sensors and actuators. *Journal of Guidance, Control and Dynamics*, **7**, 546-553.

[22] CHIANG, R. Y. and SAFONOV, M. G., 1992, Real K_m-synthesis via generalized Popov multipliers. *Proceedings of the American Control Conference*, Chicago, IL, pp. 2417-2418.

[23] CHIANG, R. Y., 1994, Parametric robustness analysis for Cassini Spacecraft using the LMI approach without frequency sweep. *Proceedings of the American Control Conference*, Baltimore, Maryland, pp. 1219-1222.

[24] COLLINS, E. G. JR., HADDAD, W. M., and DAVIS, L. D., 1994b Riccati equation approaches for small gain, positivity, and Popov robustness analysis. *Journal of Guidance, Control and Dynamics*, **17**, 322-329.

[25] COLLINS, E. G. JR., DAVIS, L. D., and RICHTER, S., 1995, Design of reduced-order, H_2 optimal controllers using a homotopy algorithm. *International Journal of Control*, **61**, 97-126.

[26] COLLINS, E. G. JR., HADDAD. W. M., and YING, S. S., 1996a, Nearly non-minimal Linear-Quadratic-Gaussian compensators for reduced-order control design initialization. *Journal of Guidance, Control and Dynamics*, **19**, 259-261.

[27] COLLINS, E. G. JR., YING, S. S., and HADDAD, W. M., 1996b, Reduced-order dynamic compensation using the Hyland and Bernstein optimal projection equations. *Journal of Guidance, Control, and Dynamics*, to appear.

[28] COLLINS, E. G. JR., HADDAD, W. M., and WATSON, L. T., 1996c, Fixed-architecture robust control design using fixed-structure multipliers. *Proceedings of the International Federation of Automatic Control*, Vol. C, pp. 73-78.

[29] COLLINS, E. G. HADDAD, W. M., WATSON, L. T., and SADHUKHAN, D., 1997a, Probability-one homotopy algorithms for robust controller synthesis with fixed-structure multipliers. *International Journal of Robust and Nonlinear Control*, **7**, 165-185.

[30] COLLINS, E. G. JR. and SADHUKHAN, D., 1997b, Synthesis of fixed-architecture, robust H_2 and $H\infty$ controllers, *Proceedings of the American Control Conference*, 67-71.

[31] COLLINS, E. G. AND SADHUKHAN, D., 1998, Robust controller synthesis via nonlinear matrix inequalities. *Proceedings of the 1998 American Control Conference*. To appear.

[32] COPLIEN, J. O., 1992, *Advanced C++, Programming Styles and Idioms*, Addison-Wesley, Redwood City, CA.

[33] DAVIS, L. D., COLLINS, E. G. JR., and HODEL, A. S., 1994, A parameterization of minimal plants. *IEEE Transactions on Automatic Control*, **39**, 849-852.

[34] DAVISON, E. J. AND KURAK E. M., 1971, A computational method for determining quadratic Lyapunov functions for non-linear systems. *Automatica*, **7**, 627-636.

[35] DENERY, D. G., 1971, An identification algorithm that is insensitive to initial parameter estimates. *AIAA Journal,* **9**, 371-377.

[36] DONGARRA, J., POZO, R., and WALKER, D., 1993, LAPACK++: A design overview of object-oriented extensions for high performance linear algebra, *Proceedings of Supercomputing '93, IEEE Press, 162-171.*

[37] DONGARRA, J., LUMSDAINE, A., POZO, R., REMINGTON, K., 1994a, A Sparse Matrix Library in C++ for High Performance Architectures, *Proceedings of the Second Annual Object-Oriented Numerics Conference,* 214-218.

[38] DONGARRA, J., LUMSDAINE, A., POZO, R., REMINGTON, K., 1994b, IML++ Iterative Methods Library Reference Guide. *http://gams.nist.gov/acmd/Staff/RPozo/sparselib++.html.*

[39] DOYLE, J. C., 1978, Guaranteed margins for LQG regulators. *IEEE Transactions on Automatic Control,* **23**, 756-757.

[40] DOYLE, J. C., and STEIN, G., 1981, Multivariable feedback design: concepts for a classical/modern synthesis. *IEEE Transactions on Automatic Control,* **26**, 4-16.

[41] DOYLE, J. C., 1982a, Analysis of feedback systems with structured uncertainties', *Proceedings of the IEE - Part D*, pp. 242-250.

[42] DOYLE, J.C., WALL, J. E., and STEIN, G., 1982b, Performance and robustness analysis for structured uncertainty. *Proceedings of the 21st IEEE Conference on Decision and Control,* Orlando, FL, pp 629-636.

[43] DOYLE, J. C., 1985, Structured uncertainty in control design. *Proceedings of the IEEE Conference on Decsion and Control,* 260–265.

[44] DOYLE, J., PRIMBS, J. A., SHAPIRO, B., and NEVISTIĆ, V., 1996, Nonlinear games: Examples and counterexamples. *Proceedings of the IEEE Conference on Decision and Control,* Kobe, Japan, 3915-3920.

[45] ERWIN, R. S., 1993, *Private communication.*

[46] FAN, M. K. H., TITS, A. L., and DOYLE, J. C., 1991, Robustness in the presence of mixed parametric uncertainty and unmodeled dynamics. *IEEE Transactions on Automatic Control,* **AC-36**, 25-38.

[47] FLETCHER, R., 1987, *Practical Methods of Optimization* (Second Edition, John Wiley and Sons, New York).

[48] FRANCIS, B. A., 1987, *A Course in H_∞ Control Theory* (Springer-Verlag, New York).

[49] FREEMAN, R. and KOKOTOVIC, P., 1996, Inverse Optimality in robust stabilization. *SIAM Journal on Optimization and Control*, **34**, 1365-1391.

[50] GAHINET, P., NEMIROVSKII, A., LAUB, A., and CHILALI, M., 1995, *LMI Control Toolbox* (The Math Works, Inc.).

[51] GE, Y., COLLINS, E. G., JR., and WATSON, L. T, 1994, A homotopy algorithm for full and reduced order mixed norm H_2/H_∞ synthesis. *Proceedings of the IEEE Conference on Decision and Control*, Orlando, FL, pp. 2672-2677. (Also submitted to *Optimal Control Applications and Methods*.)

[52] GEIST, G. A., 1991, Reduction of a general matrix to tridiagonal form. *SIAM Journal of Matrix Analysis and Applications*, **12**, 362-373.

[53] GEIST, S., *et al.*, 1993, *PVM 3 User's Guide and Reference Manual*, Oak Ridge National Laboratory.

[54] GOH, K. C., LY, J. H., TURAN, L., and SAFONOV, M. G., 1994a, μ/K_m synthesis via bilinear matrix inequalities. *Proceedings of the IEEE Conference on Decision and Control*, Orlando, FL, pp. 2032-2037.

[55] GOH, K. C., SAFONOV, M. G., and PAPAVASSILOPOULUS, G. P., 1994b, Global optimization approaches for the biaffine matrix inequality problem. *Proceedings of the IEEE Conference on Decision and Control*, Orlando, FL, pp. 2009-2014.

[56] GOH, K. C., TURAN, L., and SAFONOV, M. G., 1994c, Biaffine matrix inequality properties and computational methods. *Proceedings of the IEEE Conference on Decision and Control*, pp. 2009-2014.

[57] GOODWIN, G. C., MIDDLETON, R. H., and POOR, H. V., 1992, High-speed digital signal processing and control. *Proceedings of the IEEE*, **80**, 240-259.

[58] GRACE, A., 1992, *Optimization Toolbox: Users Guide* (The Mathworks Inc.)

[59] HADDAD, W. M., and BERNSTEIN, D. S., 1990, Generalized Riccati equations for the full- and reduced-order mixed-norm H_2/H_∞ standard problem. *Systems and Control Letters*, **14**, 185-197.

[60] HADDAD, W. M. and BERNSTEIN, D. S., 1991, Parameter-dependent Lyapunov functions, constant real parameter uncertainty, and the Popov criterion in robust analysis and synthesis, Part 1, Part 2. *Proceedings of the IEEE Conference on Decision and Control*, Brighton, U.K., pp. 2274-2279, pp. 2632-2633.

[61] HADDAD, W. M., HOW, J. P., HALL, S. R., and BERNSTEIN, D. S., 1992, Extensions of mixed-μ bounds to monotonic and odd monotonic nonlinearities using absolute stability theory, Part 1, Part 2. *Proceedings of the IEEE Conference on Decision and Control*, Tuscon, AZ, pp. 2813-2819, pp. 2820-2823.

[62] HADDAD, W. M. and BERNSTEIN, D. S., 1993, Explicit construction of quadratic Lyapunov functions for the small gain, positivity , circle, and Popov theorems and their application to robust stability, Part I: Continuous-Time Theory. *International Journal of Robust and Nonlinear Control*, **3**, 313-339.

[63] HADDAD, W. M., COLLINS, E. G. JR., and MOSER, R., 1993g, Fixed structure computation of the structured singular value. Proceedings of the American Control Conference, San Francisco, CA, 1673-1674.

[64] HADDAD, W. M., COLLINS, E. G., JR., and MOSER, R., 1994a, Structured singular value controller synthesis using constant D-scales without D-K iteration. *Proceedings of the American Control Conference*, Baltimore, MD, pp. 2819-2823. (Also in *International Journal of Control*, 1996, **63**, 813-830.)

[65] HADDAD, W. M., COLLINS, E. G., JR., and MOSER, R., 1994b, Complex structured singular value analysis using fixed structure dynamic D-scales', *Proceedings of the IEEE Conference on Decision and Control*, Orlando, FL, pp. 3003-3008. (Also to appear in *Mathematical Modeling of Systems*.)

[66] HADDAD, W. M., HOW, J. P., HALL, S. R., and BERNSTEIN, D. S., 1994c, Extensions of mixed-μ bounds to monotonic and odd monotonic nonlinearities using absolute stability theory. *International Journal of Control*, **60**, 905-951.

[67] HADDAD, W. M. and BERNSTEIN, D. S., 1995a, Parameter-dependent Lyapunov functions and the Popov criterion in robust analysis and synthesis. *IEEE Transactions on Automatic Control*, **AC-40**, 536-543.

[68] HADDAD, W. M., BERNSTEIN, D. S., and CHELLABOINA, V. S., 1995b, Generalized mixed-μ bounds for real and complex multiple-block uncertainty with internal matrix structure. *Proceedings of the American Control Conference*, Seattle, WA, pp. 2843-2847.

[69] HADDAD, W. M. and KAPILA, V., 1995af, Discrete-time extensions of mixed-μ bounds to monotonic and odd monotonic nonlinearities. *International Journal of Control*, **61**, 423-441.

[70] HADDAD, W. M., BERNSTEIN, D. S., and CHELLABOINA, V. S., 1996, Generalized mixed-μ bounds for real and complex multiple-block uncertainty with internal matrix structure. *International Journal of Control*, **64**, 789-806.

[71] HADDAD, W. M., KAPILA V., AND CHELLABOINA V. S., 1997, Guaranteed domains of attraction for multivariable Luré systems via open Lyapunov surfaces. *International Journal of Robust and Nonlinear Control*, **7**, 935-949.

[72] HADDAD, W. M., CHELLABOINA, V., and FAUSZ, J. L., 1998, Robust nonlinear feedback control for uncertain linear systems with nonquadratic performance criteria. *Systems and Control Letters*, to appear.

[73] HAHN, W., 1967, *Stability of Motion* (Berlin, DE, Springer Verlag).

[74] HENSON M. A. and SEBORG, D. E., 1996, *Nonlinear Process Control* (Prentice Hall).

[75] HITZ, L. and ANDERSON, B. D. O., 1969, Discrete positive-real functions and their applications to system stability. *Proceedings of the IEEE*, **118**, 153-155.

[76] HOW, J. P. and HALL, S. R., 1993, Connections between the Popov stability criterion and bounds for real parameter uncertainty', *Proceedings of the American Control Conference*, San Francisco, CA, pp. 1084-1089.

[77] HOROWITZ, I.,1963, *Synthesis of Feedback Systems* (Academic Press, New York).

[78] HOW, J. P., HADDAD, W. M., and HALL, S. R., 1994a, Application of Popov controller synthesis to benchmark problems with real parameter uncertainty. *AIAA Journal of Guidance, Control and Dynamics*, **17**, 759-768.

[79] HOW, J. P., HALL, S. R., and HADDAD, W. M., 1994b, Robust controllers for the middeck active control experiment using Popov controller synthesis', *IEEE Transactions on Control Systems Technology*, **2**, 73-87.

[80] HOW, J. P., COLLINS, E. G. JR., and HADDAD, W. M., 1996, Optimal Popov controller analysis and synthesis for systems with real parameter uncertainty. *IEEE Transactions on Control Systems Technology*, **4**, 200-207.

[81] Hyland, D. C. and Bernstein, D. S., 1984, The optimal projection equations for fixed-order dynamic compensation. *IEEE Transactions on Automatic Control*, **29**, 1034-1037.

[82] IWASAKI, T., 1993, *A unified matrix inequality approach to linear control design*, Ph.D. Dissertation, Purdue University.

[83] JENKINS, H. E., KURFESS, T. R., and DORF, R. C., 1996, Design of a robust controller for a grinding system. *IEEE Transactions on Control Systems Technology*, **4**, pp. 40-49.

[84] KAILATH, T., 1980, *Linear Systems* (Prentice-Hall, New Jersey).

[85] KALMAN, R. E., 1960, A new approach to linear filtering and prediction problems. *Journal of Basic Engineering, Transactions of the ASME*, **82 D**, 35-45.

[86] KALMAN, R. E., 1964, When is a linear control system optimal. *Journal of Basic Engineering, Transactions of the ASME*, **86 D**, 51-60.

[87] KALMAN, R. E., and BUCY, R. S., 1961, New results in linear filtering and prediction theory. *Journal of Basic Engineering, Transactions of the ASME*, **83 D**, 95-108.

[88] KAPILA, V. and HADDAD, W. M., 1996, A multivariable extension of the Tsypkin criterion using a Lyapunov function approach. *IEEE Transactions on Automatic Control*, **41**, 149-152.

[89] KHALIL, H. K., 1996, *Nonlinear Systems* (Prentice Hall, Upper Saddle River, NJ).

[90] KOU, R. S., ELLIOT, D. L., and TARN, T. J., 1975, Exponential observers for nonlinear dynamic systems. *Information and Control*, **29**, 204-216.

[91] KRAMER, F. S., and CALISE, A. J., 1987, Fixed-order dynamic compensation for multivariable linear systems. *Journal of Guidance and Control*, **11**, 80-85.

[92] KUHN, U., and SCHMIDT, G., 1987, Fresh look into the design and computation of optimal output feedback controls for linear multivariable systems. *International Journal of Control*, **46**, 75-95.

[93] KWAKERNAAK, H., and SIVAN, R., 1972, *Linear Optimal Control Systems* (Wiley-Interscience).

[94] LASALLE J. AND LEFSCHETZ S., 1961, *Stability by Lyapunov's Direct Method with Applications*, 58-59 (Academic Press, New York).

[95] LAUB, A., 1979, A Schur method for solving algebraic Riccati equations. *IEEE Transactions on Automatic Control*, **AC-24**, 913-921.

[96] LIU, Y., ANDERSON, B. D. O., and LY, U. L., 1990, Coprime factorization controller reduction with Bezout identity induced frequency weighting. *Automatica*, **26**, 233-249.

[97] LOPARO, K. AND BLANKENSHIP G. L., 1978, Estimating the domain of attraction of nonlinear feedback systems. *IEEE Transactions on Automatic Control*, **23**, 602-608.

[98] LY, U. L., BRYSON, A. E., and CANNON, R. H., 1985, Design of low-order compensators using parameter optimization. *Automatica*, **21**, 315-318.

[99] LY, J. H., SAFONOV, M. G., and CHIANG, R. Y., 1994, Real/Complex multivariable stability margin computation via generalized Popov multiplier - LMI approach. *Proceedings of the American Control Conference*, Baltimore, MD, pp. 425-429.

[100] LUENBERGER, D. G., 1984, *Linear and Nonlinear Programming*, 2nd Edition (Addison Wesley).

[101] MACKIEJOWSKI, J. M., 1989, *Multivariable Feedback Design*, (New York, Addison-Wesley).

[102] MAHMOUD, N. A. and KHALIL, H. 1996, Asymptotic regulation of Minimum phase nonlinear systems using output feedback *IEEE Transactions on Automatic Control*, **41**, 1402-1412.

[103] MÄKILÄ, P. M., and TOIVONEN, T., 1987, Computational methods for parametric LQ problems - a survey. *IEEE Transactionson on Automatic Control*, **32**, 658-671.

[104] MARINO, R. and TOMEI, P., 1995, *Nonlinear Control Design* (Prentice Hall).

[105] MARTIN, G. D., and BRYSON, A. E., JR., 1980, Attitude control of a flexible spacecraft. *Journal of Guidance, Control and Dynamics*, **3**, 37-41.

[106] MERCADAL, M., 1991, Homotopy approach to optimal, linear quadratic, fixed architecture compensation. *Journal of Guidance, Control and Dynamics*, **14**, 1224-1233.

[107] MICHEL, A. N., MILLER R. K. AND NAM B. H., 1982, Stability analysis of interconnected systems using computer generated Lyapunov functions. *IEEE Transactions on Circuits and Systems*, **29**, 431-440.

[108] MIDDLETON, R. H. and GOODWIN, G. C., 1986, Improved finite word length characteristics in digital control using delta operators. *IEEE Transactions on Automatic Control*, **31**, 1015-1021.

[109] MIDDLETON, R. H. and GOODWIN, G. C., 1990, *Digital Control and Estimation: A Unified Approach*, Prentice Hall, Englewood Cliffs, NJ.

[110] MOYLAN, P. and ANDERSON, B. D. O., 1973, Nonlinear regulator theory and an inverse optimal control problem. *IEEE Transactions on Automatic Control*, **18**, 460-465.

[111] MUKHOPADHYAY, V., NEWSOM, J. R., and ABEL, I., 1982, Reduced-order optimal feedback control law for flutter suppression. *Journal of Guidance and Control*, **5**, 389-395.

[112] MUKHOPADHYAY, V., 1987, Stability robustness improvement using constrained optimization techniques. *Journal of Guidance and Control*, **10**, 172-177.

[113] MUKHOPADHYAY, V., 1989, Digital robust control law synthesis using constrained optimization. *Journal of Guidance and Control*, **12**, 175-181.

[114] MUSSER, D. R., and SAINI, A., 1996, *C++ Programming with the Standard Template Library* Addison-Wesley Professional Computing Series, Reading, MA.

[115] NEMIROVSKII, A. and GAHINET, P., 1994, The projective method for solving linear matrix inequalities, *Proceedings of the American Control Conference*, pp. 840-844.

[116] NESTEROV, Y., and NEMIROVSKY, A., 1994, Interior-point polynomial methods in convex programming, 1994, *Studies in Applied Mathematics*, **13**, Philadelphia, PA.

[117] NETT, C. N., JACOBSON, C. A., and BALAS, M. J., 1984, A connection between state-space and doubly coprime fractional representations. *IEEE Transactions on Automatic Control*, **AC-29**, 831-832.

[118] NYQUIST, H., 1932, Regeneration theory. *Bell. Sys. Tech. J.*, **11**, 126-147.

[119] DE OLIVEIRA, M. C., and GEROMEL, J. C., 1997, Numerical comparison of output feedback design methods. *American Control Conference*, **1**, pp. 72-76.

[120] PACKARD, A. and DOYLE, J. C., 1993, The complex structured singular value. *Automatica*, **29**, 71-109.

[121] PARLETT, B. N., 1992, Reduction to tridiagonal form and minimal realization. *SIAM Journal of Matrix Analysis and Applications*, **12**, 567-593.

[122] PONTRYAGIN, L. S., BOLTYANSKII, V. G., GAMKRELIDZE, R. V., and MISCHENKO, E. F., 1962, *The Mathematical Theory of Optimal Process* (Interscience, New York).

[123] POZO, R., REMINGTON, K. A., and LUMSDAINE, A., 1994, SparseLib++ v. 1.3, Reference Guide, *http://gams.nist.gov/acmd/Staff/RPozo/sparselib++.html.*

[124] SAFONOV, M. G., and ATHANS, M., 1977, Gain and phase margins for multloop LQG regulators. *IEEE Transactions on Automatic Control*, **22**, 173-179.

[125] SAFONOV, M. G., 1980, *Stability and robustness of multivariable feedback systems* (MIT Press, Cambridge, MA).

[126] SAFONOV, M. G. and ATHANS, M., 1981, A multiloop generalization of the circle criterion for stability margin analysis. *IEEE Transactions on Automatic Control*, **AC-26**, 415-422.

[127] SAFONOV, M. G., 1982, Stability margins of diagonally perturbed multivariable systems. *Proceedings of the IEE*, **129-D**, 251-256.

[128] SAFONOV, M. G. and CHIANG, R. Y., 1993, Real/Complex μ without curve fitting', *Control and Dynamic Systems*, **56**, New York: Academic Press, 303-324.

[129] SAFONOV, M. G., GOH, K. C., and LY, J. H., 1994, Control system synthesis via bilinear matrix inequalities. *Proceedings of the American Control Conference*, Baltimore, MD, pp. 45-49.

[130] SAFONOV, M. G., FAN, M. K. H., 1997, Editorial. *International Journal of Robust and Nonlinear Control*, **7**, pp. 97-103.

[131] SEPULCHRE, R., JANKOVIC M., and KOKOTOVIC P., 1997, *Constructive Nonlinear Control* (Springer).

[132] SKOGESTAD, S. and POSTLETHWAITE, I., 1996, *Multivariable Feedback Control* (John Wiley and Sons).

[133] SLOTINE, J. E. and LI, W., 1991, *Applied Nonlinear Control* (Prentice Hall, Englewood Cliffs, New Jersey).

[134] STEIN. G and ATHANS, M., 1987, The LQG/LTR procedure for multivariable feedback control design. *IEEE Transactions on Automatic Control*, **32**, 105-114.

[135] STEPANOV, A., and LEE, M., 1993, The Standard Template Library, *http://www.cs.rpi.edu/projects /STL/stl-new/stl-new.html*.

[136] STROUSTRUP, B., 1991, *The C++ Programming Language, 2nd ed.* Addison-Wesley, Reading, MA.

[137] TARANTO, G. N. CHOW, J. H., and OTHMAN, H. A., 1995, Robust redesign of power system damping controllers. *IEEE Transactions on Control Systems Technology*, **3**, pp. 290-298.

[138] TCHERNYCHEV, A. and SIDERIS, A., 1996, A multiplier approach for the robust design of discrete-time control systems with mixed real/complex uncertainties. *Proceedings of the IEEE Conference on Decision and Control*, Kobe, Japan, 3484-3489.

[139] THAU, F. E., 1973, Observing the state of non-linear dynamic systems. *International Journal of Control*, **3**, 471-479.

[140] TOIVONEN, H. T. and MÄKILÄ, P. M., 1987, On Newton's method for solving parametric linear quadratic control problems. *International Journal of Control*, **21**, pp. 743-744.

[141] TSYPKIN, Y. Z., 1964, A criterion for absolute stability of automatic pulse systems with monotonic characteristics of the nonlinear element. *Sov. Phys. Doklady*, **9**, 263-266.

[142] VANDENBERGHE, L., and BOYD, S., 1996a, Semidefinite Programming, *SIAM Review*, **38**, 49-95.

[143] VANDENBERGHE, L, and BOYD, S., 1996b, *Software for Semidefinite Programming, User's Guide.* preprint.

[144] VANNELLI, A. and VIDYASAGAR M., 1985, Maximal Lyapunov functions and domains of attraction for autonomous nonlinear systems. *Automatica*, **21**, 69-80.

[145] VIDYASAGAR, M., 1985, *Control System Synthesis: A Factorization Approach* (Cambridge, MA: MIT Press).

[146] VIDYASAGAR, M., 1993, *Nonlinear System Analysis* (New Jersey: Prentice Hall).

[147] DE VILLEMAGNE, C., and SKELTON, R. E., 1988, Controller reduction using canonical interactions. *IEEE Transactions on Automatic Control*, **33**, 740-750.

[148] VOTH, C., and LY, U. L., 1991, Design of a total energy control autopilot using constrained parameter optimization. *Journal of Guidance, Control and Dynamics*, **14**, 927-935.

[149] WATSON, L. T., 1987a, Numerical linear algebra aspects of globally convergent homotopy methods. *SIAM Review*, **28**, 529-545.

[150] WATSON, L. T., BILLUPS, S. C., and MORGAN, A. P., 1987b, Algorithm 652: HOMPACK: A suite of codes for globally convergent homotopy algorithms. *ACM Transactions on Mathematical Software*, **13**, 281-310.

[151] WEI, B. and BERNSTEIN, D. S., 1992, Benchmark problems for robust control design. *Journal of Guidance, Control, and Dynamics*, **15**, 1057-1059.

[152] WEN, J. T., 1988, Time domain and frequency domain conditions for strict positive realness', *IEEE Transactions on Automatic Control*, **3**, 988-992.

[153] Whorton, M., Buschek, H., and Calise, A. J., 1994, Homotopy algorithm for fixed-order H_2 and H_∞ design, *Proceedings of the AIAA Guidance, Navigation and Control Conference*, 988-992.

[154] WONG, P. K., 1975, *On the interaction structure of linear multi-input feedback control systems* Masters Thesis, MIT, 988-992.

[155] YING, S., 1993, *Reduced-order H_2 modeling and control using the optimal projection equations: theoretical issues and computational algorithms*, Ph.D. Dissertation, Florida Institute of Technology, Melbourne, Florida.

[156] YOUNG, P. M.,1993, *Robustness with parametric and dynamic uncertainty*, Ph.D. Dissertation, California Institute of Technology, Pasadena, CA.

[157] YOUNG, P. M., 1996, Controller design with real parametric uncertainty. *International Journal of Control*, **65**, 469-509.

[158] YOUSUFF, A. and SKELTON, R. E., 1984a, A note on balanced controller reduction. *IEEE Transactions on Automatic Control*, **29**, 254-257.

[159] YOUSUFF, A. and SKELTON, R. E., 1984b, Controller reduction by controller cost analysis. *IEEE Transactions on Automatic Control*, **29**, 520-530.

[160] ZAHEER-UDDIN, M., PATEL, R. V., and AL-ASSADI, S. A. K., 1993, Design of decentralized robust controllers for multizone space heating systems. *IEEE Transactions on Control Systems Technology*, **1**, 246-261.

[161] ZAMES, G., 1981, Feedback and optimal sensitivity: model reference transformations, multiplicative seminorms and approximate inverses. *IEEE Transactions on Automatic Control*, **26**, 301-320.

[162] ZUBOV, V. I., 1964, *Methods of A. M. Lyapunov and their applications*, Groninger, NL, P. Noordhoff.